

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平7-152499

(43)公開日 平成7年(1995)6月16日

(51)Int.Cl.⁶

G 0 6 F 3/06

識別記号

庁内整理番号

F I

技術表示箇所

5 4 0

3 0 5 C

審査請求 未請求 請求項の数44 FD (全31頁)

(21)出願番号 特願平6-209197

(22)出願日 平成6年(1994)8月10日

(31)優先権主張番号 特願平5-273200

(32)優先日 平5(1993)10月5日

(33)優先権主張国 日本 (JP)

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 角田 仁

東京都国分寺市東恋ヶ窪1丁目280番地

株式会社日立製作所中央研究所内

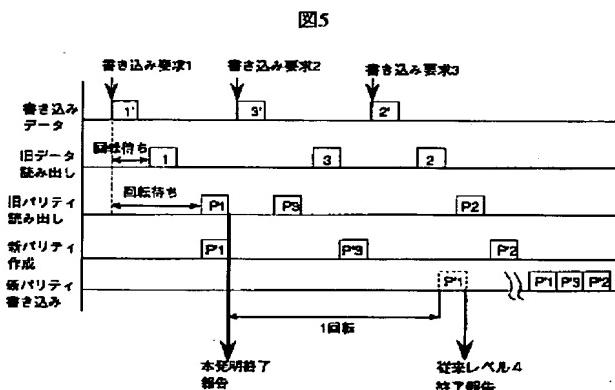
(74)代理人 弁理士 笹岡 茂 (外1名)

(54)【発明の名称】ディスクアレイ装置におけるパリティ格納方法、障害回復方法およびディスクアレイ装置

(57)【要約】

【目的】RAID(レベル4、5)のディスクアレイシステムにおいて、データの書き込みによるパリティの更新時のオーバヘッドを減少させる。

【構成】書き込み要求1、3、2が上位装置から送られ、書き込みデータ1'、3'、2'が送られると、旧データ1、3、2および旧パリティP1、P2、P3を夫々回転待ちの後読み出し、書き込みデータと旧データと旧パリティにより新パリティをP'1、P'2、P'3を作成し、キャッシングメモリに貯蔵しておき、新パリティが予めユーザが設定した値以上になった場合か、読み出し書き込み要求の発行されていないタイミングが生じた場合にパリティ格納用のドライブにまとめて書き込む。上位装置への書き込み処理終了報告は、新パリティをキャッシングメモリに格納した時点で行なわれる。パリティ格納用のドライブに替えてパリティ格納用のフラッシュメモリを用いることもできる。



【特許請求の範囲】

【請求項1】 上位装置に接続され、キャッシュメモリと少なくとも1台の制御装置を備えるディスクアレイコントローラと、該ディスクアレイコントローラに接続され複数台のディスク装置からなる少なくとも1つの論理グループを備えるディスクアレイユニットとを備え、前記ディスクアレイコントローラの制御装置が、前記上位装置から1回に読み出しありは書き込みする単位で転送されてきたデータを分割せずに前記複数台のディスク装置内の複数台のデータ用のディスク装置のいずれかに格納し、前記複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、前記複数台のディスク装置の内のパリティ用のディスク装置に格納するディスクアレイ装置におけるパリティ格納方法であって、

このディスクアレイ装置に対し上位装置から発行された書き込み要求に対し、前記制御装置はパリティを更新した後、この更新したパリティを前記キャッシュメモリに格納し、同様に上位装置から発行されてきた別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵し、このキャッシュメモリ内の更新されたパリティ群を書き込み要求の発行順に前記パリティ用のディスク装置の更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項2】 請求項1記載のディスクアレイ装置におけるパリティ格納方法において、

前記ディスクアレイコントローラに更新されたパリティを格納する専用メモリを設け、更新されたパリティを前記キャッシュメモリに替えて前記専用メモリに格納するようにしたことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項3】 請求項2記載のディスクアレイ装置におけるパリティ格納方法において、

前記専用メモリを揮発メモリとしたことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項4】 請求項1記載のディスクアレイ装置におけるパリティ格納方法において、

前記キャッシュメモリにアドレス変換用テーブルを設け、

該テーブルには、パリティグループの各データの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスと、パリティの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスとパリティをキャッシュメモリ内に貯蔵した場合のキャッシュアドレスとキャッシュメモリ内に貯蔵されたパリティとディスク装置内に格納されているパリティの何れかを示すフラグを備えることを特徴とするディスクアレイ装置におけるパリティ

格納方法。

【請求項5】 請求項4記載のディスクアレイ装置におけるパリティ格納方法において、

更新されたパリティ群をシーケンシャルにまとめてディスク装置に書き込む途中において、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれているディスク装置のアドレスにおいては、一旦書き込みを中断し、書き込み要求が発行され、すでに更新されているディスク装置の無効なパリティのアドレスにおいて、シーケンシャル書き込みを再開することを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項6】 請求項4記載のディスクアレイ装置におけるパリティ格納方法において、

更新されたパリティ群をシーケンシャルにまとめてディスク装置に書き込む際、ディスク装置の書き込み先の領域に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれている場合、この有効なパリティをキャッシュメモリに読み出し、更新されたパリティと合わせてパリティ群を構成すると共にパリティの書き込み順を決定し、該パリティ群を該書き込み順にディスク装置の書き込み前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項7】 請求項1記載のディスクアレイ装置におけるパリティ格納方法において、

前記パリティ用のディスク装置を複数台設け、更新されたパリティ群をまとめてディスク装置に書き込む際、複数のパリティ用のディスク装置に、更新されたパリティ単位に、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項8】 請求項1記載のディスクアレイ装置におけるパリティ格納方法において、

前記パリティ用のディスク装置を複数台設け、更新されたパリティ群をまとめてディスク装置に書き込む際、複数のパリティ用のディスク装置に、バイト単位に、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項9】 請求項1記載のディスクアレイ装置におけるパリティ格納方法において、

更新されたパリティ群をシーケンシャルにまとめて書き込む前記パリティ用のディスクを複数の領域に分割し、更新される前のパリティが所属する領域ごとに、更新されたパリティ群を作成し、該作成されたパリティ群を該パリティ群が所属する領域に、かつ書き込み要求の発行順に更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項10】 上位装置に接続され、キャッシュメモリと少なくとも1台の制御装置を備えるディスクアレイコントローラと、該ディスクアレイコントローラに接続され複数台のディスク装置からなる少なくとも1つの論理グループを備えるディスクアレイユニットとを備え、前記ディスクアレイコントローラの制御装置が、前記上位装置から1回に読み出しされた書込みする単位で転送されてきたデータを分割せずに前記複数台のディスク装置のいずれかに格納し、前記複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、前記複数台のディスク装置のいずれかに格納するディスクアレイ装置におけるパリティ格納方法であって、

前記複数台のディスク装置の夫々にパリティ格納領域を設け、該パリティ格納領域が割り当てられるディスク装置内の領域を各ディスク装置において夫々異なるディスク装置内の領域とし、

このディスクアレイ装置に対し上位装置から発行された書き込み要求に対し、前記制御装置はパリティを更新した後、この更新したパリティを前記キャッシュメモリに格納し、同様に上位装置から発行されてきた別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵し、このキャッシュメモリ内の更新されたパリティ群を更新される前のパリティが所属するディスク装置のパリティ格納領域毎に作成し、対応するディスク装置のパリティ格納領域に、該パリティ群を書き込み要求の発行順に更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項11】 請求項1乃至請求項3のいずれかの請求項記載のディスクアレイ装置におけるパリティ格納方法において、

前記パリティ用のディスク装置をフラッシュメモリとし、該パリティ用のフラッシュメモリ上の前記更新前のパリティ群を消去した後、前記キャッシュメモリ内の更新されたパリティ群を書き込み要求の発行順に前記パリティ用のフラッシュメモリの消去された更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項12】 請求項11記載のディスクアレイ装置におけるパリティ格納方法において、

前記キャッシュメモリにアドレス変換用テーブルを設け、

該テーブルには、パリティグループの各データの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスと、パリティの論理アドレスと該論理アドレスに対応するフラッシュメモリチップ番号およびフラッシュメモリチップ内アドレスとパ

リティをキャッシュメモリ内に貯蔵した場合のキャッシュアドレスとキャッシュメモリ内に貯蔵されたパリティとディスク装置内に格納されているパリティの何れが有効かを示すフラグを備えることを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項13】 請求項12記載のディスクアレイ装置におけるパリティ格納方法において、

更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む時の、実際にパリティをフラッ

10 シュメモリに書き込む前に、フラッシュメモリ内のパリティを書き込むアドレスの消去を行う際に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれているフラッシュメモリのアドレスにおいては消去を行わずに、書き込み要求が発行され、すでに更新されているフラッシュメモリの無効なパリティのアドレスに対してのみ、消去を行うことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項14】 請求項13記載のディスクアレイ装置におけるパリティ格納方法において、

20 更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む際に、書き込み要求が発行され、すでに消去されているフラッシュメモリの無効なパリティのアドレスに対してのみ、書き込みを行うことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項15】 請求項12記載のディスクアレイ装置におけるパリティ格納方法において、

更新されたパリティ群をシーケンシャルにまとめてフラッ

30 シュメモリに書き込む際、フラッシュメモリの更新されたパリティ群の書き込み先の領域に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれている場合、この有効なパリティをキャッシュメモリに読み出し、更新されたパリティと合わせてパリティ群を構成すると共にパリティの書き込み順を決定し、該パリティ群に対応するフラッシュメモリ上の書き込み前のパリティ群を消去した後、前記キャッシュメモリ内のパリティ群を前記書き込み順にフラッシュメモリの書き込み前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むことを特徴とするディスクア

40 レイ装置におけるパリティ格納方法。

【請求項16】 請求項12または請求項15記載のディスクアレイ装置におけるパリティ格納方法において、

更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む際、フラッシュメモリ内の低いアドレスから高いアドレスに向かって順次書き込んでいく、最も高いアドレスまでパリティを書き込んだら、最も低いアドレスに戻り、又、順次パリティを書き込んでいく、リングバッファのようにフラッシュメモリにパリティをシーケンシャルに書き込んでいくことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項17】 請求項16記載のディスクアレイ装置におけるパリティ格納方法において、
フラッシュメモリ内の最も低いアドレスにパリティを書き込んだ回数のカウントをすることを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項18】 請求項17記載のディスクアレイ装置におけるパリティ格納方法において、
前記カウントした回数に基づきフラッシュメモリの寿命を判定し、寿命がきたことを出力することを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項19】 請求項11記載のディスクアレイ装置におけるパリティ格納方法において、
前記パリティ用のフラッシュメモリを複数のフラッシュメモリチップで構成し、更新されたパリティ群をまとめてフラッシュメモリに書き込む際、複数のフラッシュメモリチップに、更新されたパリティ単位に、更新前のパリティを消去した後、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項20】 請求項11記載のディスクアレイ装置におけるパリティ格納方法において、
前記パリティ用のフラッシュメモリを複数のフラッシュメモリチップで構成し、更新されたパリティ群をまとめてフラッシュメモリに書き込む際、複数のフラッシュメモリチップに、バイト単位に、更新前のパリティを消去した後、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込むことを特徴とするディスクアレイ装置におけるパリティ格納方法。

【請求項21】 上位装置に接続され、キャッシュメモリと少なくとも1台の制御装置を備えるディスクアレイコントローラと、該ディスクアレイコントローラに接続され複数台のディスク装置と複数個のフラッシュメモリからなる少なくとも1つの論理グループを備えるディスクアレイユニットとを備え、
前記ディスクアレイコントローラの制御装置が、前記上位装置から1回に読み出しありは書き込みする単位で転送されてきたデータを分割せずに前記複数台のディスク装置の内の複数台のデータ用のディスク装置のいずれかに格納し、前記複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、前記複数台のディスク装置の内のパリティ用のディスク装置に格納するディスクアレイ装置において、

前記ディスクアレイコントローラの制御装置が、前記上位装置から1回に読み出しありは書き込みする単位で転送されてきたデータを分割せずに前記複数台のディスク装置の内の複数台のデータ用のディスク装置のいずれかに格納し、前記複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、前記複数台のディスク装置の内のパリティ用のディスク装置に格納するディスクアレイ装置において、

前記制御装置は、このディスクアレイ装置に対し上位装置から発行された書き込み要求に対し、パリティを更新した後、この更新したパリティを前記キャッシュメモリに格納し、同様に上位装置から発行されてきた別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵する手段と、このキャッシュメモリ内の

更新されたパリティ群を、該パリティ群に対応するフラッシュメモリ上の更新前のパリティ群を消去した後、前記キャッシュメモリ内の更新されたパリティ群を書き込み要求の発行順に前記パリティ用のフラッシュメモリの消去された更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込まれたディスクアレイ装置において、データが格納されているドライブに障害が発生した場合は、このドライブに格納されているデータを復するたびに、パリティが格納されているフラッシュメモリから、当該パリティを読み出すことを特徴とするディスクアレイにおける障害回復方法。

【請求項22】 上位装置に接続され、キャッシュメモリと少なくとも1台の制御装置を備えるディスクアレイコントローラと、該ディスクアレイコントローラに接続され複数台のディスク装置からなる少なくとも1つの論理グループを備えるディスクアレイユニットとを備え、前記ディスクアレイコントローラの制御装置が、前記上位装置から1回に読み出しありは書き込みする単位で転送されてきたデータを分割せずに前記複数台のディスク装置の内の複数台のデータ用のディスク装置のいずれかに格納し、前記複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、前記複数台のディスク装置の内のパリティ用のディスク装置に格納するディスクアレイ装置において、

前記制御装置は、このディスクアレイ装置に対し上位装置から発行された書き込み要求に対し、パリティを更新した後、この更新したパリティを前記キャッシュメモリに格納し、同様に上位装置から発行されてきた別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵する手段と、このキャッシュメモリ内の更新されたパリティ群を書き込み要求の発行順に前記パリティ用のディスク装置の更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項23】 請求項22記載のディスクアレイ装置において、

前記ディスクアレイコントローラに更新されたパリティを格納する専用メモリを設け、更新されたパリティを前記キャッシュメモリに替えて前記専用メモリに格納するようにしたことを特徴とするディスクアレイ装置。

【請求項24】 請求項23記載のディスクアレイ装置において、

前記専用メモリを揮発メモリとしたことを特徴とするディスクアレイ装置。

【請求項25】 請求項22記載のディスクアレイ装置において、

前記キャッシュメモリにアドレス変換用テーブルを設け、

該テーブルには、パリティグループの各データの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスと、パリティの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスとパリティをキャッシュメモリ内に貯蔵した場合のキャッシュアドレスとキャッシュメモリ内に貯蔵されたパリティとディスク装置内に格納されているパリティの何れが有効かを示すフラグを備えることを特徴とするディスクアレイ装置。

【請求項26】 請求項25記載のディスクアレイ装置において、

前記制御装置は、更新されたパリティ群をシーケンシャルにまとめてディスク装置に書き込む途中において、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれているディスク装置のアドレスにおいては、一旦書き込みを中断し、書き込み要求が発行され、すでに更新されているディスク装置の無効なパリティのアドレスにおいて、シーケンシャル書き込みを再開する手段を備えることを特徴とするディスクアレイ装置。

【請求項27】 請求項25記載のディスクアレイ装置におけるパリティ格納方法において、

前記制御装置は、更新されたパリティ群をシーケンシャルにまとめてディスク装置に書き込む際、ディスク装置の書き込み先の領域に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれている場合、この有効なパリティをキャッシュメモリに読み出し、更新されたパリティと合わせてパリティ群を構成すると共にパリティの書き込み順を決定する手段と、該パリティ群を該書き込み順にディスク装置の書き込み前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項28】 請求項25記載のディスクアレイ装置において、

前記アドレス変換用テーブルには、不必要になったために削除されたデータであることを示す無効フラグを前記各データ対応に備えることを特徴とするディスクアレイ装置。

【請求項29】 請求項28記載のディスクアレイ装置において、

前記制御装置は、ディスク装置に新規にデータを書き込む際、ディスク装置に未使用な領域がない場合に、前記無効フラグが立っているデータの領域にデータを書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項30】 請求項22記載のディスクアレイ装置において、

前記パリティ用のディスク装置を複数台設け、前記制御装置は、更新されたパリティ群をまとめてディスク装置に書き込む際、複数のパリティ用のディスク装置に、更

新されたパリティ単位に、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項31】 請求項22記載のディスクアレイ装置において、

前記パリティ用のディスク装置を複数台設け、前記制御装置は、更新されたパリティ群をまとめてディスク装置に書き込む際、複数のパリティ用のディスク装置に、バイト単位に、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項32】 請求項22記載のディスクアレイ装置において、

更新されたパリティ群をシーケンシャルにまとめて書き込む前記パリティ用のディスクを複数の領域に分割し、前記制御装置は、更新される前のパリティが所属する領域ごとに、更新されたパリティ群を作成する手段と、該作成されたパリティ群を該パリティ群が所属する領域に、かつ書き込み要求の発行順に更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項33】 上位装置に接続され、キャッシュメモリと少なくとも1台の制御装置を備えるディスクアレイコントローラと、該ディスクアレイコントローラに接続され複数台のディスク装置からなる少なくとも1つの論理グループを備えるディスクアレイユニットとを備え、前記ディスクアレイコントローラの制御装置が、前記上位装置から1回に読み出しちゃは書込みする単位で転送されてきたデータを分割せずに前記複数台のディスク装置のいずれかに格納し、前記複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、前記複数台のディスク装置のいずれかに格納するディスクアレイ装置において、前記複数台のディスク装置の夫々にパリティ格納領域を設け、該パリティ格納領域が割り当てるディスク装置内の領域を各ディスク装置において夫々異なるディスク装置内の領域とし、

前記制御装置は、このディスクアレイ装置に対し上位装置から発行された書き込み要求に対し、パリティを更新した後、この更新したパリティを前記キャッシュメモリに格納し、同様に上位装置から発行されてきた別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵し、このキャッシュメモリ内の更新されたパリティ群を更新される前のパリティが所属するディスク装置のパリティ格納領域毎に作成する手段と、該パリティ群を対応するディスク装置のパリティ格納領域に、かつ書き込み要求の発行順に更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込む手段を備えることを特徴とするディスクアレイ装置。

前記制御装置は、このディスクアレイ装置に対し上位装置から発行された書き込み要求に対し、パリティを更新した後、この更新したパリティを前記キャッシュメモリに格納し、同様に上位装置から発行されてきた別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵し、このキャッシュメモリ内の更新されたパリティ群を更新される前のパリティが所属するディスク装置のパリティ格納領域毎に作成する手段と、該パリティ群を対応するディスク装置のパリティ格納領域に、かつ書き込み要求の発行順に更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項34】 請求項22乃至請求項24のいずれかの請求項記載のディスクアレイ装置において、

前記パリティ用のディスク装置をフラッシュメモリとし、該パリティ用のフラッシュメモリ上の前記更新前のパリティ群を消去した後、前記キャッシュメモリ内の更新されたパリティ群を書き込み要求の発行順に前記パリティ用のフラッシュメモリの消去された更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項35】 請求項34記載のディスクアレイ装置において、

前記キャッシュメモリにアドレス変換用テーブルを設け、

該テーブルには、パリティグループの各データの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスと、パリティの論理アドレスと該論理アドレスに対応するフラッシュメモリチップ番号およびフラッシュメモリチップ内アドレスとパリティをキャッシュメモリ内に貯蔵した場合のキャッシュアドレスとキャッシュメモリ内に貯蔵されたパリティとフラッシュメモリ内に格納されているパリティの何れが有効かを示すフラグを備えることを特徴とするディスクアレイ装置。

【請求項36】 請求項35記載のディスクアレイ装置において、

前期制御装置は、更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む時の、実際にパリティをフラッシュメモリに書き込む前に、フラッシュメモリ内のパリティを書き込むアドレスの消去を行う際に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれているフラッシュメモリのアドレスにおいては消去を行わずに、書き込み要求が発行され、すでに更新されているフラッシュメモリの無効なパリティのアドレスに対してのみ、消去を行う手段を備えることを特徴とするディスクアレイ装置。

【請求項37】 請求項36記載のディスクアレイ装置において、

更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む際に、書き込み要求が発行され、すでに消去されているフラッシュメモリの無効なパリティのアドレスに対してのみ、書き込みを行う手段を備えることを特徴とするディスクアレイ装置。

【請求項38】 請求項35記載のディスクアレイ装置において、

前記制御装置は、更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む際、フラッシュメモリの更新されたパリティ群の書き込み先の領域に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれている場合、この有効なパリティをキャッシュメモリに読み出し、更新された

パリティと合わせてパリティ群を構成すると共にパリティの書き込み順を決定手段と、該パリティ群に対応するフラッシュメモリ上の書き込み前のパリティ群を消去する手段と、前記キャッシュメモリ内のパリティ群を前記書き込み順にフラッシュメモリの書き込み前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項39】 請求項35または請求項38記載のディスクアレイ装置において、

10 更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む際、フラッシュメモリ内の低いアドレスから高いアドレスに向かって順次書き込んでいく手段と、最も高いアドレスまでパリティを書き込んだら、最も低いアドレスに戻り、又、順次パリティを書き込んでいく、リングバッファのようにフラッシュメモリにパリティをシーケンシャルに書き込んでいく手段を備えることを特徴とするディスクアレイ装置。

【請求項40】 請求項39記載のディスクアレイ装置において、

20 フラッシュメモリ内の最も低いアドレスにパリティを書き込んだ回数のカウントをする手段を備えることを特徴とするディスクアレイ装置。

【請求項41】 請求項40記載のディスクアレイ装置において、

前記カウントした回数に基づきフラッシュメモリの寿命を判定し、寿命がきたことを出力する手段を備えることを特徴とするディスクアレイ装置。

【請求項42】 請求項34記載のディスクアレイ装置において、

30 前記パリティ用のフラッシュメモリチップを複数台設け、前記制御装置は、更新されたパリティ群をまとめてフラッシュメモリに書き込む際、複数のフラッシュメモリチップに、更新されたパリティ単位に、更新前のパリティを消去した後、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項43】 請求項34記載のディスクアレイ装置において、

前記パリティ用のフラッシュメモリチップを複数台設け、前記制御装置は、更新されたパリティ群をまとめてフラッシュメモリに書き込む際、複数のフラッシュメモリチップに、バイト単位に、更新前のパリティを消去した後、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込む手段を備えることを特徴とするディスクアレイ装置。

【請求項44】 上位装置に接続され、キャッシュメモリと少なくとも1台の制御装置を備えるディスクアレイコントローラと、該ディスクアレイコントローラに接続され複数台のディスク装置と複数個のフラッシュメモリからなる少なくとも1つの論理グループを備えるディス

クアレイユニットとを備え、

前記ディスクアレイコントローラの制御装置が、前記上位装置から1回に読み出しありは書き込みする単位で転送されてきたデータを分割せずに前記複数台のディスク装置内の複数台のデータ用のディスク装置のいずれかに格納し、前記複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、前記フラッシュメモリに格納するディスクアレイ装置において、

前記制御装置は、このディスクアレイ装置に対し上位装置から発行された書き込み要求に対し、パリティを更新した後、この更新したパリティを前記キャッシュメモリに格納し、同様に上位装置から発行されてきた別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵する手段と、このキャッシュメモリ内の更新されたパリティ群を、一度にシーケンシャルに前記パリティ用のフラッシュメモリに書き込まれたディスクアレイ装置において、データが格納されているドライブに障害が発生した場合は、このドライブに格納されているデータを回復するたびに、パリティが格納されているフラッシュメモリから、当該パリティを読み出す手段を備えることを特徴とするディスクアレイ装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明はディスクファイルシステムに係り、特に高性能な入出力動作を可能とするディスクアレイ装置と該装置におけるパリティ格納方法および障害回復方法に関する。

【0002】

【従来の技術】現在のコンピュータシステムにおいては、CPU等の上位側が必要とするデータは2次記憶装置に格納され、CPUが必要とする時に応じ2次記憶装置に対してデータの書き込み、読み出しを行っている。この2次記憶装置としては一般に不揮発な記憶媒体が使用され、代表的なものとして磁気ディスク装置（以下ドライブとする）、光ディスクなどがあげられる。近年高度情報化に伴い、コンピュータシステムにおいて、2次記憶装置の高性能化が要求してきた。その一つの解として、多数の比較的容量の小さなドライブにより構成されるディスクアレイが考えられている。ディスクアレイについて記載された文献として、「D. Patterson, G. Gibson, and R. H. Kartz; A Case for Redundant Arrays of Inexpensive Disks (RAID), in ACM SIGMOD Conference, Chicago, IL, (June 1988)」がある。D. Patterson, G. Gibson, and R. H. Kartz; A Case for Redundant Arrays of In

expensive Disks (RAID)において、データを分割して並列に処理を行うディスクアレイ（レベル3）とデータを分散して、独立に扱うディスクアレイ（レベル4、5）について、その性能および信頼性の検討結果が報告されている。現在この論文に書かれている方式が最も一般的なディスクアレイと考えられている。

【0003】レベル4、5のディスクアレイでは個々のデータを分割せずに独立に扱い、多数の比較的容量の小さなドライブに分散して格納するものである。以下にデータを分散して、独立に扱うディスクアレイについて説明する。レベル4はレベル5において論理グループを構成するドライブに分散しているパリティを、1台のパリティのみを格納するドライブにまとめたものである。ここで、レベル3、レベル4、レベル5について簡単に説明しておく。レベル3は、ディスクに格納するデータ#1として、例えば「001010101011...」を想定し、データ#1とパリティを格納するためのディスクとしてディスク#1～#5が設けられた場合、

ディスク#1に‘0’、ディスク#2に‘0’、ディスク#3に‘1’、ディスク#4に‘0’を順次格納し、格納された‘0010’に対するパリティをディスク#5に格納する。そして、次に同様にして‘1’、‘0’、‘1’、‘0’を順次ディスク#1～#4に格納し、そのパリティを#5に格納してゆく。レベル4は、データとパリティを格納するためのディスクとしてディスク#1～#5が設けられた場合、データ#1、#5、...がディスク#1に、データ#2、#6、...がディスク#2に、データ#3、#7、...がディスク#3に、データ#4、#8、...がディスク#4に格納される。そして、例えば、データ#1が「01...」、データ#2が「00...」、データ#3が「11...」、データ#4が「00...」であるとすると、各データの先頭ビット‘0010’に対するパリティをパリティ専用として指定されたディスク#5の先頭ビットととして格納し、以下同様にして、各データの2番目のビット‘1010’に対するパリティをディスク#5の2番目のビットととして格納してゆく。そして、データ#5～#8のデータ組に対するパリティデータをディスク#5に2番目のパリティデータとして格納するようにしてゆく。レベル5は、レベル4のようなパリティ専用のディスクを決めず、データ#1をディスク#1、データ#2をディスク#2、データ#3をディスク#3、データ#4をディスク#4に格納し、データ#1～#4のデータ組に対するパリティデータP1234をディスク#5に格納し、次いで、データ#5をディスク#2、データ#6をディスク#3、データ#7をディスク#4、データ#8をディスク#5に格納し、データ#5～#8のデータ組に対するパリティデータP5678をディスク#1に格納し、次いで、データ#9をディスク#1、

ディスク#1に格納し、次いで、データ#10をディスク#2、データ#11をディスク#3、データ#12をディスク#4、データ#13をディスク#5に格納し、次いで、データ#14をディスク#6、データ#15をディスク#7、データ#16をディスク#8、データ#17をディスク#9に格納し、次いで、データ#18をディスク#10、データ#19をディスク#11、データ#20をディスク#12、データ#21をディスク#13に格納し、次いで、データ#22をディスク#14、データ#23をディスク#15、データ#24をディスク#16、データ#25をディスク#17に格納し、次いで、データ#26をディスク#18、データ#27をディスク#19、データ#28をディスク#20、データ#29をディスク#21に格納し、次いで、データ#30をディスク#22、データ#31をディスク#23、データ#32をディスク#24、データ#33をディスク#25に格納し、次いで、データ#34をディスク#26、データ#35をディスク#27、データ#36をディスク#28、データ#37をディスク#29に格納し、次いで、データ#38をディスク#30、データ#39をディスク#31、データ#40をディスク#32、データ#41をディスク#33に格納し、次いで、データ#42をディスク#34、データ#43をディスク#35、データ#44をディスク#36、データ#45をディスク#37に格納し、次いで、データ#46をディスク#38、データ#47をディスク#39、データ#48をディスク#40、データ#49をディスク#41に格納し、次いで、データ#50をディスク#42、データ#51をディスク#43、データ#52をディスク#44、データ#53をディスク#45に格納し、次いで、データ#54をディスク#46、データ#55をディスク#47、データ#56をディスク#48、データ#57をディスク#49に格納し、次いで、データ#58をディスク#50、データ#59をディスク#51、データ#60をディスク#52、データ#61をディスク#53に格納し、次いで、データ#62をディスク#54、データ#63をディスク#55、データ#64をディスク#56、データ#65をディスク#57に格納し、次いで、データ#66をディスク#58、データ#67をディスク#59、データ#68をディスク#60、データ#69をディスク#61に格納し、次いで、データ#70をディスク#62、データ#71をディスク#63、データ#72をディスク#64、データ#73をディスク#65に格納し、次いで、データ#74をディスク#66、データ#75をディスク#67、データ#76をディスク#68、データ#77をディスク#69に格納し、次いで、データ#78をディスク#70、データ#79をディスク#71、データ#80をディスク#72、データ#81をディスク#73に格納し、次いで、データ#82をディスク#74、データ#83をディスク#75、データ#84をディスク#76、データ#85をディスク#77に格納し、次いで、データ#86をディスク#78、データ#87をディスク#79、データ#88をディスク#80、データ#89をディスク#81に格納し、次いで、データ#90をディスク#82、データ#91をディスク#83、データ#92をディスク#84、データ#93をディスク#85に格納し、次いで、データ#94をディスク#86、データ#95をディスク#87、データ#96をディスク#88、データ#97をディスク#89に格納し、次いで、データ#98をディスク#90、データ#99をディスク#91、データ#100をディスク#92、データ#101をディスク#93に格納し、次いで、データ#102をディスク#94、データ#103をディスク#95、データ#104をディスク#96、データ#105をディスク#97に格納し、次いで、データ#106をディスク#98、データ#107をディスク#99、データ#108をディスク#100、データ#109をディスク#101に格納し、次いで、データ#110をディスク#102、データ#111をディスク#103、データ#112をディスク#104、データ#113をディスク#105に格納し、次いで、データ#114をディスク#106、データ#115をディスク#107、データ#116をディスク#108、データ#117をディスク#109に格納し、次いで、データ#118をディスク#110、データ#119をディスク#111、データ#120をディスク#112、データ#121をディスク#113に格納し、次いで、データ#122をディスク#114、データ#123をディスク#115、データ#124をディスク#116、データ#125をディスク#117に格納し、次いで、データ#126をディスク#118、データ#127をディスク#119、データ#128をディスク#120、データ#129をディスク#121に格納し、次いで、データ#130をディスク#122、データ#131をディスク#123、データ#132をディスク#124、データ#133をディスク#125に格納し、次いで、データ#134をディスク#126、データ#135をディスク#127、データ#136をディスク#128、データ#137をディスク#129に格納し、次いで、データ#138をディスク#130、データ#139をディスク#131、データ#140をディスク#132、データ#141をディスク#133に格納し、次いで、データ#142をディスク#134、データ#143をディスク#135、データ#144をディスク#136、データ#145をディスク#137に格納し、次いで、データ#146をディスク#138、データ#147をディスク#139、データ#148をディスク#140、データ#149をディスク#141に格納し、次いで、データ#150をディスク#142、データ#151をディスク#143、データ#152をディスク#144、データ#153をディスク#145に格納し、次いで、データ#154をディスク#146、データ#155をディスク#147、データ#156をディスク#148、データ#157をディスク#149に格納し、次いで、データ#158をディスク#150、データ#159をディスク#151、データ#160をディスク#152、データ#161をディスク#153に格納し、次いで、データ#162をディスク#154、データ#163をディスク#155、データ#164をディスク#156、データ#165をディスク#157に格納し、次いで、データ#166をディスク#158、データ#167をディスク#159、データ#168をディスク#160、データ#169をディスク#161に格納し、次いで、データ#170をディスク#162、データ#171をディスク#163、データ#172をディスク#164、データ#173をディスク#165に格納し、次いで、データ#174をディスク#166、データ#175をディスク#167、データ#176をディスク#168、データ#177をディスク#169に格納し、次いで、データ#178をディスク#170、データ#179をディスク#171、データ#180をディスク#172、データ#181をディスク#173に格納し、次いで、データ#182をディスク#174、データ#183をディスク#175、データ#184をディスク#176、データ#185をディスク#177に格納し、次いで、データ#186をディスク#178、データ#187をディスク#179、データ#188をディスク#180、データ#189をディスク#181に格納し、次いで、データ#190をディスク#182、データ#191をディスク#183、データ#192をディスク#184、データ#193をディスク#185に格納し、次いで、データ#194をディスク#186、データ#195をディスク#187、データ#196をディスク#188、データ#197をディスク#189に格納し、次いで、データ#198をディスク#190、データ#199をディスク#191、データ#200をディスク#192、データ#201をディスク#193に格納し、次いで、データ#202をディスク#194、データ#203をディスク#195、データ#204をディスク#196、データ#205をディスク#197に格納し、次いで、データ#206をディスク#198、データ#207をディスク#199、データ#208をディスク#200、データ#209をディスク#201に格納し、次いで、データ#210をディスク#202、データ#211をディスク#203、データ#212をディスク#204、データ#213をディスク#205に格納し、次いで、データ#214をディスク#206、データ#215をディスク#207、データ#216をディスク#208、データ#217をディスク#209に格納し、次いで、データ#218をディスク#210、データ#219をディスク#211、データ#220をディスク#212、データ#221をディスク#213に格納し、次いで、データ#222をディスク#214、データ#223をディスク#215、データ#224をディスク#216、データ#225をディスク#217に格納し、次いで、データ#226をディスク#218、データ#227をディスク#219、データ#228をディスク#220、データ#229をディスク#221に格納し、次いで、データ#230をディスク#222、データ#231をディスク#223、データ#232をディスク#224、データ#233をディスク#225に格納し、次いで、データ#234をディスク#226、データ#235をディスク#227、データ#236をディスク#228、データ#237をディスク#229に格納し、次いで、データ#238をディスク#230、データ#239をディスク#231、データ#240をディスク#232、データ#241をディスク#233に格納し、次いで、データ#242をディスク#234、データ#243をディスク#235、データ#244をディスク#236、データ#245をディスク#237に格納し、次いで、データ#246をディスク#240、データ#247をディスク#241、データ#248をディスク#242、データ#249をディスク#243に格納し、次いで、データ#250をディスク#244、データ#251をディスク#245、データ#252をディスク#246、データ#253をディスク#247に格納し、次いで、データ#254をディスク#248、データ#255をディスク#249、データ#256をディスク#250、データ#257をディスク#251に格納し、次いで、データ#258をディスク#252、データ#259をディスク#253、データ#260をディスク#254、データ#261をディスク#255に格納し、次いで、データ#262をディスク#256、データ#263をディスク#257、データ#264をディスク#258、データ#265をディスク#259に格納し、次いで、データ#266をディスク#260、データ#267をディスク#268、データ#268をディスク#269、データ#269をディスク#270に格納し、次いで、データ#271をディスク#271、データ#272をディスク#272、データ#273をディスク#273、データ#274をディスク#274に格納し、次いで、データ#275をディスク#275、データ#276をディスク#276、データ#277をディスク#277、データ#278をディスク#278に格納し、次いで、データ#279をディスク#279、データ#280をディスク#280、データ#281をディスク#281、データ#282をディスク#282に格納し、次いで、データ#283をディスク#283、データ#284をディスク#284、データ#285をディスク#285、データ#286をディスク#286に格納し、次いで、データ#287をディスク#287、データ#288をディスク#288、データ#289をディスク#289、データ#290をディスク#290に格納し、次いで、データ#291をディスク#291、データ#292をディスク#292、データ#293をディスク#293、データ#294をディスク#294に格納し、次いで、データ#295をディスク#295、データ#296をディスク#296、データ#297をディスク#297、データ#298をディスク#298に格納し、次いで、データ#299をディスク#299、データ#300をディスク#300、データ#301をディスク#301、データ#302をディスク#302に格納し、次いで、データ#303をディスク#303、データ#304をディスク#304、データ#305をディスク#305、データ#306をディスク#306に格納し、次いで、データ#307をディスク#307、データ#308をディスク#308、データ#309をディスク#309、データ#310をディスク#310に格納し、次いで、データ#311をディスク#311、データ#312をディスク#312、データ#313をディスク#313、データ#314をディスク#314に格納し、次いで、データ#315をディスク#315、データ#316をディスク#316、データ#317をディスク#317、データ#318をディスク#318に格納し、次いで、データ#319をディスク#319、データ#320をディスク#320、データ#321をディスク#321、データ#322をディスク#322に格納し、次いで、データ#323をディスク#323、データ#324をディスク#324、データ#325をディスク#325、データ#326をディスク#326に格納し、次いで、データ#327をディスク#327、データ#328をディスク#328、データ#329をディスク#329、データ#330をディスク#330に格納し、次いで、データ#331をディスク#331、データ#332をディスク#332、データ#333をディスク#333、データ#334をディスク#334に格納し、次いで、データ#335をディスク#335、データ#336をディスク#336、データ#337をディスク#337、データ#338をディスク#338に格納し、次いで、データ#339をディスク#339、データ#340をディスク#340、データ#341をディスク#341、データ#342をディスク#342に格納し、次いで、データ#343をディスク#343、データ#344をディスク#344、データ#345をディスク#345、データ#346をディスク#346に格納し、次いで、データ#347をディスク#347、データ#348をディスク#348、データ#349をディスク#349、データ#350をディスク#350に格納し、次いで、データ#351をディスク#351、データ#352をディスク#352、データ#353をディスク#353、データ#354をディスク#354に格納し、次いで、データ#355をディスク#355、データ#356をディスク#356、データ#357をディスク#357、データ#358をディスク#358に格納し、次いで、データ#359をディスク#359、データ#360をディスク#360、データ#361をディスク#361、データ#362をディスク#362に格納し、次いで、データ#363をディスク#363、データ#364をディスク#364、データ#365をディスク#365、データ#366をディスク#366に格納し、次いで、データ#367をディスク#367、データ#368をディスク#368、データ#369をディスク#369、データ#370をディスク#370に格納し、次いで、データ#371をディスク#371、データ#372をディスク#372、データ#373をディスク#373、データ#374をディスク#374に格納し、次いで、データ#375をディスク#375、データ#376をディスク#376、データ#377をディスク#377、データ#378をディスク#378に格納し、次いで、データ#379をディスク#379、データ#380をディスク#380、データ#381をディスク#381、データ#382をディスク#382に格納し、次いで、データ#383をディスク#383、データ#384をディスク#384、データ#385をディスク#385、データ#386をディスク#386に格納し、次いで、データ#387をディスク#387、データ#388をディスク#388、データ#389をディスク#389、データ#390をディスク#390に格納し、次いで、データ#391をディスク#391、データ#392をディスク#392、データ#393をディスク#393、データ#394をディスク#394に格納し、次いで、データ#395をディスク#395、データ#396をディスク#396、データ#397をディスク#397、データ#398をディスク#398に格納し、次いで、データ#399をディスク#399、データ#400をディスク#400、データ#401をディスク#401、データ#402をディスク#402に格納し、次いで、データ#403をディスク#403、データ#404をディスク#404、データ#405をディスク#405、データ#406をディスク#406に格納し、次いで、データ#407をディスク#407、データ#408をディスク#408、データ#409をディスク#409、データ#410をディスク#410に格納し、次いで、データ#411をディスク#411、データ#412をディスク#412、データ#413をディスク#413、データ#414をディスク#414に格納し、次いで、データ#415をディスク#415、データ#416をディスク#416、データ#417をディスク#417、データ#418をディスク#418に格納し、次いで、データ#419をディスク#419、データ#420をディスク#420、データ#421をディスク#421、データ#422をディスク#422に格納し、次いで、データ#423をディスク#423、データ#424をディスク#424、データ#425をディスク#425、データ#426をディスク#426に格納し、次いで、データ#427をディスク#427、データ#428をディスク#428、データ#429をディスク#429、データ#430をディスク#430に格納し、次いで、データ#431をディスク#431、データ#432をディスク#432、データ#433をディスク#433、データ#434をディスク#434に格納し、次いで、データ#435をディスク#435、データ#436をディスク#436、データ#437をディスク#437、データ#438をディスク#438に格納し、次いで、データ#439をディスク#439、データ#440をディスク#440、データ#441をディスク#441、データ#442をディスク#442に格納し、次いで、データ#443をディスク#443、データ#444をディスク#444、データ#445をディスク#445、データ#446をディスク#446に格納し、次いで、データ#447をディスク#447、データ#448をディスク#448、データ#449をディスク#449、データ#450をディスク#450に格納し、次いで、データ#451をディスク#451、データ#452をディスク#452、データ#453をディスク#453、データ#454をディスク#454に格納し、次いで、データ#455をディスク#455、データ#456をディスク#456、データ#457をディスク#457、データ#458をディスク#458に格納し、次いで、データ#459をディスク#459、データ#460をディスク#460、データ#461をディスク#461、データ#462をディスク#462に格納し、次いで、データ#463をディスク#463、データ#464をディスク#464、データ#465をディスク#465、データ#466をディスク#466に格納し、次いで、データ#467をディスク#467、データ#468をディスク#468、データ#469をディスク#469、データ#470をディスク#470に格納し、次いで、データ#471をディスク#471、データ#472をディスク#472、データ#473をディスク#473、データ#474をディスク#474に格納し、次いで、データ#475をディスク#475、データ#476をディスク#476、データ#477をディスク#477、データ#478をディスク#478に格納し、次いで、データ#479をディスク#479、データ#480をディスク#480、データ#481をディスク#481、データ#482をディスク#482に格納し、次いで、データ#483をディスク#483、データ#484をディスク#484、データ#485をディスク#485、データ#486をディスク#486に格納し、次いで、データ#487をディスク#487、データ#488をディスク#488、データ#489をディスク#489、データ#490をディスク#490に格納し、次いで、データ#491をディスク#491、データ#492をディスク#492、データ#493をディスク#493、データ#494をディスク#494に格納し、次いで、データ#495をディスク#495、データ#496をディスク#496、データ#497をディスク#497、データ#498をディスク#498に格納し、次いで、データ#499をディスク#499、データ#500をディスク#500、データ#501をディスク#501、データ#502をディスク#502に格納し、次いで、データ#503をディスク#503、データ#504をディスク#504、データ#505をディスク#505、データ#506をディスク#506に格納し、次いで、データ#507をディスク#507、データ#508をディスク#508、データ#509をディスク#509、データ#510をディスク#510に格納し、次いで、データ#511をディスク#511、データ#512をディスク#512、データ#513をディスク#513、データ#514をディスク#514に格納し、次いで、データ#515をディスク#515、データ#516をディスク#516、データ#517をディスク#517、データ#518をディスク#518に格納し、次いで、データ#519をディスク#519、データ#520をディスク#520、データ#521をディスク#521、データ#522をディスク#522に格納し、次いで、データ#523をディスク#523、データ#524をディスク#524、データ#525をディスク#525、データ#526をディスク#526に格納し、次いで、データ#527をディスク#527、データ#528をディスク#528、データ#529をディスク#529、データ#530をディスク#530に格納し、次いで、データ#531をディスク#531、データ#532をディスク#532、データ#533をディスク#533、データ#534をディスク#534に格納し、次いで、データ#535をディスク#535、データ#536をディスク#536、データ#537をディスク#537、データ#538をディスク#538に格納し、次いで、データ#539をディスク#539、データ#540をディスク#540、データ#541をディスク#541、データ#542をディスク#542に格納し、次いで、データ#543をディスク#543、データ#544をディスク#544、データ#545をディスク#545、データ#546をディスク#546に格納し、次いで、データ#547をディスク#547、データ#548をディスク#548、データ#549をディスク#549、データ#550をディスク#550に格納し、次いで、データ#551をディスク#551、データ#552をディスク#552、データ#553をディスク#553、データ#554をディスク#554に格納し、次いで、データ#555をディスク#555、データ#556をディスク#556、データ#557をディスク#557、データ#558をディスク#558に格納し、次いで、データ#559をディスク#559、データ#560をディスク#560、データ#561をディスク#561、データ#562をディスク#562に格納し、次いで、データ#563をディスク#563、データ#564をディスク#564、データ#565をディスク#565、データ#566をディスク#566に格納し、次いで、データ#567をディスク#567、データ#568をディスク#568、データ#569をディスク#569、データ#570をディスク#570に格納し、次いで、データ#571をディスク#571、データ#572をディスク#572、データ#573をディスク#573、データ#574をディスク#574に格納し、次いで、データ#575をディスク#575、データ#576をディスク#576、データ#577をディスク#577、データ#578をディスク#578に格納し、次いで、データ#579をディスク#579、データ#580をディスク#580、データ#581をディスク#581、データ#582をディスク#582に格納し、次いで、データ#583をディスク#583、データ#584をディスク#584、データ#585をディスク#585、データ#586をディスク#586に格納し、次いで、データ#587をディスク#587、データ#588をディスク#588、データ#589をディスク#589、データ#590をディスク#590に格納し、次いで、データ#591をディスク#591、データ#592をディスク#592、データ#593をディスク#593、データ#594をディスク#594に格納し、次いで、データ#595をディスク#595、データ#596をディスク#596、データ#597をディスク#597、データ#598をディスク#598に格納し、次いで、データ#599をディスク#599、データ#600をディスク#600、データ#601をディスク#601、データ#602をディスク#602に格納し、次いで、データ#603をディスク#603、データ#604をディスク#604、データ#605をディスク#605、データ#606をディスク#606に格納し、次いで、データ#607をディスク#607、データ#608をディスク#608、データ#609をディスク#609、データ#610をディスク#610に格納し、次いで、データ#611をディスク#611、データ#612をディスク#612、データ#613をディスク#613、データ#614をディスク#614に格納し、次いで、データ#615をディスク#615、データ#616をディスク#616、データ#617をディスク#617、データ#618をディスク#618に格納し、次いで、データ#619をディスク#619、データ#620をディスク#620、データ#621をディスク#621、データ#622をディスク#622に格納し、次いで、データ#623をディスク#623、データ#624をディスク#624、データ#625をディスク#625、データ#626をディスク#626に格納し、次いで、データ#627をディスク#627、データ#628をディスク#628、データ#629をディスク#629、データ#630をディスク#630に格納し、次いで、データ#631をディスク#631、データ#632をディスク#632、データ#633をディスク#633、データ#634をディスク#634に格納し、次いで、データ#635をディスク#635、データ#636をディスク#636、データ#637をディスク#637、データ#638をディスク#638に格納し、次いで、データ#639をディスク#639、データ#640をディスク#640、データ#641をディスク#641、データ#642をディスク#642に格納し、次いで、データ#643をディスク#643、データ#644をディスク#644、データ#645をディスク#645、データ#646をディスク#646に格納し、次いで、データ#647をディスク#647、データ#648をディスク#648、データ#649をディスク#649、データ#650をディスク#650に格納し、次いで、データ#651をディスク#651、データ#652をディスク#652、データ#653をディスク#653、データ#654

データ#10をディスク#3、データ#11をディスク#4、データ12をディスク#5に格納し、データ#9～#12のデータ組に対するパリティデータP9101112をディスク#2に格納するようにしてゆく。

【0004】現在、一般に使用されている汎用大型コンピュータシステムの2次記憶装置では、1ドライブ当たりの容量が大きいため、他の読み出し／書き込み要求に当該ドライブが使用されて、そのドライブを使用できずに待たされることが多く発生した。上記文献に記載されたタイプのディスクアレイでは汎用大型コンピュータシステムの2次記憶装置で使用されている大容量のドライブを、多数の比較的容量の小さなドライブで構成し、データを分散して格納してあるため、読み出し／書き込み要求が増加してもディスクアレイの複数のドライブで分散して処理することが可能となり、読み出し／書き込み要求がまたされることが減少する。しかし、ディスクアレイは、このように多数のドライブにより構成されるため、部品点数が増加し障害が発生する確率が高くなる。そこで、信頼性の向上を図る目的で、パリティを用意する必要がある。このパリティによりデータを格納したドライブに障害が発生した場合、その障害ドライブ内のデータを復元することが可能となる。ディスクアレイではデータからパリティを作成しデータと同様にドライブに格納しておく。この時、パリティは、パリティの作成に関与したデータとは別のドライブに格納される。

【0005】これらのディスクアレイでは、現在一般に使用されている汎用大型コンピュータシステムと同様、2次記憶装置内では、個々のデータの格納場所（アドレス）は予め指定したアドレスに固定され、CPUから当該データへ読み出しありは書き込みする場合、この固定されたアドレスへアクセスすることになっている。この分散して格納するディスクアレイ（レベル5）ではストレージテクノロジコーポレーション（以下STKとする）から製品発表がされている。STK社から出願されている米国特許WO 91/20076では、レベル5の基本アーキテクチャにおいて、動的に変更可能なアドレスのテーブルを用意することにより、データ圧縮を行いデータの書き込み処理において、トラック単位で書き込み先のアドレスを動的に変換する方法について開示されている。また、IBM社の特開平4-230512号公報にはレベル5において、書き込み時に書き込むデータと、この書き込みにより更新したパリティを、それぞれ別の場所に書き込む方法について開示されている。さらに、IBM社からディスクアレイ（9337）では、レベル5においてWAD（ライト アシスト デバイス）を設けることが発表されている。

【0006】一方、近年磁気ディスクの置き換えデバイスとしてフラッシュメモリが着目されている。フラッシュメモリは不揮発な半導体メモリのため、磁気ディスクと比較して高速にデータの読み出し、書き込みが可能で

ある。しかし、フラッシュメモリでは書き込む際に書き込み先に書き込まれているデータを消去してからでなければ書き込めない。HN28F1600シリーズのフラッシュメモリのデータシート（ADJ-203-045（A）（z））によると、データの書き込みまたは読みだし時間は、DRAM等と同様に約100ns程度だが、消去時間が10msかかる。また、フラッシュメモリでは書き込み回数に限界があり、一般にフラッシュメモリでは百万回が書き込み回数の限界とされ実用化においては問題とされている。このように、フラッシュメモリにおける、書き込み回数に限界があるという問題点を解決する方法として、書き込み時にマッピングテーブルでフラッシュメモリへの書き込み回数が平均化するようアドレス変換する方法についてIBM社から特開平5-27924号公報において開示されている。

【0007】

【発明が解決しようとする課題】現在の汎用大型計算機システム等ではドライブにより構成される2次記憶装置内では、CPUから転送されてくるデータは個々のデータの格納場所（アドレス）が予め指定したアドレスに固定され、CPUから当該データへ読み出しありは書き込む場合は、この固定されたアドレスへアクセスすることになる。これは、ディスクアレイにおいても同じである。データを分割して並列に処理を行うディスクアレイ（レベル3）ではこのようにアドレスを固定しても影響はないが、データを分散して、独立に扱うディスクアレイ（レベル4、5）ではアドレスを固定した場合、書き込み時に大きな処理オーバヘッドが必要になる。この書き込み時の処理オーバヘッドについては特願平4-230512に説明されている。以下それについて説明する。

【0008】図10は公知例で示したD. Pattersonらが提案したRAIDに述べられている、データを分散して独立に扱うディスクアレイ（レベル5）内部のデータアドレスを示している。この各アドレスにあるデータは1回の読み出し／書き込み処理される単位で、個々のデータは独立している。前述したようにこのようなシステムでは、信頼性を向上するためパリティを設定することが不可欠である。本システムでは各ドライブ内の同一アドレスのデータによりパリティが作成される。すなわち、ドライブ#1から4までのアドレス（1, 1）のデータの組によりパリティが作成され、パリティを格納するドライブの（1, 1）に格納される。本システムでは読み出し／書き込み処理は現在の汎用大型計算機システムと同様に各ドライブに対し当該データをアクセスする。このようなディスクアレイにおいて、例えばドライブ#3のアドレス（2, 2）のデータを更新する場合、

まず、更新される前のドライブ#3の（2, 2）のデータと、パリティを格納してあるドライブの（2, 2）のパリティを読み出す（1）。読み出したデータと、読み出したパリティと、更新する新しいデータとの排他的論

理和をとり、新たなパリティを作成する(2)。パリティの作成完了後、更新する新しいデータをドライブ#3の(2, 2)に、新パリティをパリティを格納するドライブの(2, 2)に格納する(3)。

【0009】このようなレベル5のディスクアレイでは、データの格納されているドライブ、パリティの格納されているドライブから古いデータとパリティを読み出すため、ディスクを平均1/2回転待ち、それから読み出してパリティを作成する。この新しく作成したパリティを書き込むため更に1回転必要となり、データを書き替える場合最低で1.5回転待たなければならない。特に更新されたパリティを書き込む際に待たされる1回転の回転待ちが、書き込み時の性能低下を引き起こす大きな問題である。このように、ドライブにおいては1.5回転ディスクの回転を待つということは非常に大きなオーバヘッドとなる。この書き込み時のオーバヘッドはレベル4においても同様である。このような書き込み時のオーバヘッドを削減するため、書き込み先のアドレスを動的に変換する方法が考えられ、STK社から出願されているWO 91/20076に開示されている。また、IBM社から出願されている特願平4-230512号公報においても、書き込み時において書き込みデータを書き込みデータが書き込まれるアドレスではなく別のアドレスに書き込むことにより書き込みオーバヘッドを削減する方法について開示されている。このように、レベル5のディスクアレイでは、読み出しと比較し書き込み時ではパリティ生成とこの生成したパリティを書き込む処理のオーバヘッドが非常に大きいため、CPUからの読み出し、書き込み要求が多いときには、この処理オーバヘッドが性能低下の大きな原因となる。

【0010】本発明の目的は、ディスクアレイにおけるパリティを書き込み処理のオーバヘッドを大幅に減少させることにある。本発明の他の目的は、ディスクアレイにおいて、パリティをフラッシュメモリ(FMEM)に格納することで、書き込み処理のオーバヘッドを大幅に減少させることにある。本発明のさらに他の目的としては、パリティを格納するFMEMの書き込み回数を平均化することにある。

【0011】

【課題を解決するための手段】上位装置から1回に読み出しましたは書き込みする単位で転送されてきたデータを分割せずに複数台のディスク装置内の複数台のデータ用のディスク装置のいずれかに格納し、複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、複数台のディスク装置内のパリティ用のディスク装置に格納するディスクアレイ装置において、このディスクアレイ装置に対し上位装置から発行された書き込み要求に対し、パリティを更新した後、この更新したパリティを前記キャッシュメモリに格納し、同様に上位装置から発行されてきた

別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵し、このキャッシュメモリ内の更新されたパリティ群を書き込み要求の発行順に前記パリティ用のディスク装置の更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むようしている。また、前記キャッシュメモリにアドレス変換用テーブルを設け、該テーブルには、パリティグループの各データの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスと、パリティの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスとパリティをキャッシュメモリ内に貯蔵した場合のキャッシュアドレスとキャッシュメモリ内に貯蔵されたパリティとディスク装置内に格納されているパリティの何れが有効かを示すフラグを備えるようしている。また、キャッシュメモリにアドレス変換用テーブルを設けておき、更新されたパリティ群をシーケンシャルにまとめてディスク装置に書き込む途中において、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれているディスク装置のアドレスにおいては、一旦書き込みを中断し、書き込み要求が発行され、すでに更新されているディスク装置の無効なパリティのアドレスにおいて、シーケンシャル書き込みを再開するようしている。また、キャッシュメモリにアドレス変換用テーブルを設けておき、更新されたパリティ群をシーケンシャルにまとめてディスク装置に書き込む際、ディスク装置の書き込み先の領域に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれている場合、この有効なパリティをキャッシュメモリに読み出し、更新されたパリティと合わせてパリティ群を構成すると共にパリティの書き込み順を決定し、該パリティ群を該書き込み順にディスク装置の書き込み前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むようしている。また、パリティ用のディスク装置を複数台設け、更新されたパリティ群をまとめてディスク装置に書き込む際、複数のパリティ用のディスク装置に、更新されたパリティ単位に、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込むようしている。更新されたパリティ群をシーケンシャルにまとめて書き込むパリティ用のディスクを複数の領域に分割し、更新される前のパリティが所属する領域ごとに、更新されたパリティ群を作成し、該作成されたパリティ群を該パリティ群が所属する領域に、かつ書き込み要求の発行順に更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むようしている。また、パリティ用のディスク装置を設けず、複数台のディスク装置の夫々にパリティ格納領域を設け、該パリティ格納領域が割り当たられるディスク装置内の領域を各ディスク装置において夫々異なるディスク装置内の

領域とし、上位装置から発行された書き込み要求に対し、パリティを更新した後、この更新したパリティをキャッシュメモリに格納し、同様に上位装置から発行されてきた別の書き込み要求に対し更新したパリティも該キャッシュメモリに格納し、これらの更新されたパリティを該キャッシュメモリ内に貯蔵し、このキャッシュメモリ内の更新されたパリティ群を更新される前のパリティが所属するディスク装置のパリティ格納領域毎に作成し、対応するディスク装置のパリティ格納領域に、該パリティ群を書き込み要求の発行順に更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むようしている。

【0012】また、上位装置から1回に読み出しあるは書き込みする単位で転送されてきたデータを分割せずに複数台のディスク装置内の複数台のデータ用のディスク装置のいずれかに格納し、複数台のデータ用のディスク装置に格納されている各データによりパリティを生成し、この生成したパリティを、複数台のディスク装置内のパリティ用のディスク装置に格納するディスクアレイ装置において、該パリティ用のディスク装置をフラッシュメモリとし、該パリティ用のフラッシュメモリ上の前記更新前のパリティ群を消去した後、前記キャッシュメモリ内の更新されたパリティ群を書き込み要求の発行順に前記パリティ用のフラッシュメモリの消去された更新前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むようしている。また、キャッシュメモリにアドレス変換用テーブルを設け、該テーブルには、パリティグループの各データの論理アドレスと該論理アドレスに対応するディスク装置番号およびディスク装置内アドレスと、パリティの論理アドレスと該論理アドレスに対応するフラッシュメモリチップ番号およびフラッシュメモリチップ内アドレスとパリティをキャッシュメモリ内に貯蔵した場合のキャッシュアドレスとキャッシュメモリ内に貯蔵されたパリティとディスク装置内に格納されているパリティの何れが有効かを示すフラグを備えるようしている。また、更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む時の、実際にパリティをフラッシュメモリに書き込む前に、フラッシュメモリ内のパリティを書き込むアドレスの消去を行う際に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれているフラッシュメモリのアドレスにおいては消去を行わずに、書き込み要求が発行され、すでに更新されているフラッシュメモリの無効なパリティのアドレスに対してのみ、消去を行うようしている。さらに、更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む際に、書き込み要求が発行され、すでに消去されているフラッシュメモリの無効なパリティのアドレスに対してのみ、書き込みを行なうようしている。また、更新されたパリティ群をシーケンシャルにまとめて

フラッシュメモリに書き込む際、フラッシュメモリの更新されたパリティ群の書き込み先の領域に、書き込み要求が発行されていないため更新されていない有効なパリティが書き込まれている場合、この有効なパリティをキャッシュメモリに読み出し、更新されたパリティと合わせてパリティ群を構成すると共にパリティの書き込み順を決定し、該パリティ群に対応するフラッシュメモリ上の書き込み前のパリティ群を消去した後、前記キャッシュメモリ内のパリティ群を前記書き込み順にフラッシュメモリの書き込み前のパリティ群の一連のアドレスへ一度にシーケンシャルに書き込むようしている。また、更新されたパリティ群をシーケンシャルにまとめてフラッシュメモリに書き込む際、フラッシュメモリ内の低いアドレスから高いアドレスに向かって順次書き込んでいく、最も高いアドレスまでパリティを書き込んだら、最も低いアドレスに戻り、又、順次パリティを書き込んでいく、リングバッファのようにフラッシュメモリにパリティをシーケンシャルに書き込んでいくようしている。また、フラッシュメモリ内の最も低いアドレスにパリティを書き込んだ回数のカウントをするようしている。また、上記カウントした回数に基づきフラッシュメモリの寿命を判定し、寿命がきたことを出力するようしている。また、パリティ用のフラッシュメモリを複数のフラッシュメモリチップで構成し、更新されたパリティ群をまとめてフラッシュメモリに書き込む際、複数のフラッシュメモリチップに、更新されたパリティ単位に、更新前のパリティを消去した後、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込むようしている。また、パリティ用のフラッシュメモリを複数のフラッシュメモリチップで構成し、更新されたパリティ群をまとめてフラッシュメモリに書き込む際、複数のフラッシュメモリチップに、バイト単位に、更新前のパリティを消去した後、書き込み要求の発行順に更新前のパリティの一連のアドレスへ一度に並列に書き込むようしている。また、データが格納されているドライブに障害が発生した場合は、このドライブに格納されているデータを回復するたびに、パリティが格納されているフラッシュメモリから、当該パリティを読み出し、障害回復をするようしている。

【0013】

【作用】データの書き込みによるパリティの更新後、更新された新パリティを更新のたびに更新前のパリティが格納されていたドライブのアドレスに書き込むと、その都度回転待ちが必要となる。本発明では、更新パリティをキャッシュメモリに貯蔵しておき、これらの更新パリティをシーケンシャルにまとめ書きすることにより、このまとめ書きを行なう時に0.5回転の回転待ちが必要となるが、まとめ書きを開始以降は回転待ちは無い。つまり、まとめられた更新パリティの集団の中で、一番最初に書き込まれるパリティについては0.5回転の回転

待ちが必要であるが、引き続き書き込まれる2番目以降のパリティについては回転待ちがなくなる。また、更新パリティをキャッシュメモリに貯蔵しておき、これらの更新パリティをシーケンシャルにFMEMにまとめ書きすることにより、このまとめ書きを行なう時は最初に一括消去の時間（約10ms）が必要となるのみである。FMEMでは部分消去時間と一括消去時間はほとんど同じである。つまり、まとめられた更新パリティの集団の中で、一番最初に書き込まれるパリティについては、全ての更新パリティに対応する更新前のパリティを一括消去する時間が必要であるが、引き続き書き込まれる2番目以降のパリティについてはFMEMへの書き込み時間のみとなる。また、FMEMに対する新パリティのシーケンシャルなまとめ書きは、アドレスの低い方から高い方へ一方向に行なわれるため、FMEMチップでは書き込み回数が平均化される。

【0014】

【実施例】

（実施例1）以下本発明の一実施例を説明する。図1は、本実施例のハードウェア構成を示し、1はCPU、2はディスクアレイコントローラ（以下ADC）、3はディスクアレイユニット（以下ADU）である。ADU3は複数の論理グループ10により構成され、個々の論理グループ10はm台のドライブ12と、各々のドライブ12とADC2を接続するディスクアレイユニットバス9-1から9-4により構成される。なお、このドライブ12の数については特に制限は無い。この論理グループ10は障害回復単位で、この論理グループ10内の各ドライブ12内の各データによりパリティを作成する。本実施例ではm-1台の個々のドライブ12内のデータから各々のパリティが作成される。

【0015】次にADC2の内部構造について図1を用いて説明する。ADC2はチャネルバスディレクタ5と2個のクラスタ13とバッテリバックアップ等により不揮発化された半導体メモリであるキャッシュメモリ7により構成される。このキャッシュメモリ7にはデータとアドレス変換用テーブルが格納されている。このキャッシュメモリ7およびその中のアドレス変換用テーブルはADC2内の全てのクラスタ13において共有で使用される。クラスタ13はADC2内において独立に動作可能なバスの集合で、各クラスタ13間においては電源、回路は全く独立となっている。クラスタ13はチャネル、キャッシュメモリ7間のバスであるチャネルバス6と、キャッシュメモリ7、ドライブ12間のバスであるドライブバス6-1から4が、それぞれ、2個ずつで構成されている。それぞれのチャネルバス6-1から4とドライブバス8はキャッシュメモリ7を介して接続されている。CPU1より発行されたコマンドは外部インターフェースバス4を通じてADC2のチャネルバスディレクタ5に発行される。ADC2は2個のクラスタ13

により構成され、それぞれのクラスタは2個のバスで構成されるため、ADC2は合計4個のバスにより構成される。このことから、ADC2ではCPU1からのコマンドを同時に4個まで受け付けることが可能である。そこで、CPU1からコマンドが発行された場合ADC2内のチャネルバスディレクタ5によりコマンドの受付が可能かどうか判断する。

【0016】図2は図1のチャネルバスディレクタ5と1クラスタ13-1内の内部構造を示した図である。図2に示すように、CPU1からADC2に送られてきたコマンドはインターフェースアダプタ（以下IF Adp）15により取り込まれ、マイクロプロセッサであるMP20はクラスタ内の外部インターフェースバス4の中で使用可能なバスがあるかを調べ、使用可能な外部インターフェースバス4がある場合はMP20はチャネルバススイッチ16を切り換えてコマンドの受け付け処理を行ない、受け付けられない場合は受付不可の応答をCPU1へ送る。

【0017】（アドレス変換法）本実施例ではADU3

を構成するドライブ12はSCSIインターフェースのドライブを使用する。以下にアドレス変換について説明する。CPU1は論理アドレスとしてデータ名を指定し、ADC2のMP20により実際のドライブ12内の物理的アドレスであるドライブ12内のアドレス(SCSI内Addr)に変換される。SCSI内Addrは図11に示すように当該データが格納されているトラックが所属するシリンドの位置とそのシリンド内において当該データが格納されているトラックを決定するヘッドアドレスと、そのトラック内のレコードの位置で構成される。具体的には要求データが格納されている当該ドライブ12の番号と当該ドライブ12内のシリンド番号であるシリンドアドレスとシリンドにおいてトラックを選択するヘッドの番号であるヘッドアドレスとレコードアドレスからなる。このアドレス変換には以下に示すようなアドレス変換用のテーブル（以下アドレステーブルとする）が使用される。ADC2内のキャッシュメモリ7には、その内部の適当な領域に図4に示すようなアドレステーブルが格納されている。

【0018】アドレステーブルは論理アドレス27に対し、無効データの場合オン(1)となる無効フラグ28と、データが格納されているドライブ12のアドレスであるデータドライブ番号29(Drive No.)と、そのドライブ12内の実際にデータが格納されている物理アドレスであるSCSI内Addr30と、そのデータがキャッシュメモリ7内にある場合のキャッシュメモリ7内のアドレスであるキャッシュアドレス31と、キャッシュメモリ7内にデータがある場合オン(1)とするキャッシュフラグ32と、パリティグループにおいてパリティの論理アドレスであるP論理アドレス33と、パリティが格納されているドライブ番号であ

るパリティドライブ番号 34 (P D r i v e N o .) と、そのドライブ 1 2 内の実際にパリティが格納されている物理アドレスである P S C S I 内 A d d r 3 5 と、パリティの格納されているキャッシュアドレスである P キャッシュアドレス 3 6 と、パリティがキャッシュメモリ 7 内に存在するか否かを示す P キャッシュフラグ 3 7 により構成されている。パリティは、論理グループ 1 0 を構成する各ドライブ 1 2 において同一 S C S I 内 A d d r 3 0 のデータにより作成される。パリティグループは、論理グループ 1 0 を構成する各ドライブ 1 2 において同一 S C S I 内 A d d r 3 0 のデータと、これらのデータから作成されたパリティにより構成される。具体的には、図 4 において S C S I 内 A d d r 3 0 が D A D R 1 については S D # 1 のドライブ 1 2 に格納されている Data # 1 と、 S D # 2 のドライブ 1 2 に格納されている Data # 2 と、 S D # 3 のドライブ 1 2 に格納されている Data # 3 と、 S D # 4 のドライブ 1 2 に格納されている Data # 4 とにより作成されたパリティである Par i t y # 1 が S D # 5 のドライブ 1 2 の P S C S I 内 A D D R が D A D R 5 に格納され、これらのデータとパリティがパリティグループを構成する。

【0019】以上に説明したアドレステーブルにより、 C P U が指定する論理アドレス 2 7 に基づき、データが実際に格納されているドライブ番号 2 9 と S C S I 内 A d d r 3 0 を決定する。例えば、図 4 において C P U 1 から Data # 2 に対し要求を発行してきた場合、アドレステーブルから S D # 2 のドライブ 1 2 内の S C S I 内 A d d r 3 0 として D A D R 1 が該当していることが分かり、物理的なアドレスへ変換される。また、この Data # 2 に対応するパリティは、 P 論理アドレス 3 3 が Par i t y # 1 で、パリティドライブ番号 (P D r i v e N o .) 3 4 が S D # 5 のドライブ 1 2 であり、 P S C S I 内 A D D R が D A D R 5 の位置に格納されている。このように、 C P U 1 から指定された論理アドレス 2 7 を、実際に読み出し／書き込みを行うドライブ 1 2 の物理的なアドレスに変換した後、 S D # 2 のドライブ 1 2 の Data # 2 に対し読み出しありは書き込み要求が発行される。この時アドレステーブルにおいて Data # 2 の論理アドレス 2 7 ではキャッシュフラグ 3 2 がオン (1) のため、このデータはキャッシュメモリ 7 内の C A D R 5 に存在する。もし、キャッシュフラグ 3 2 がオフ (0) であればキャッシュメモリ 7 内には、当該データは存在しない。この、アドレステーブルはシステムの電源をオンした時に、 M P 2 0 により論理グループ 1 0 内のある特定のドライブ 1 2 から、キャッシュメモリ 7 に C P U 1 の閲知無しに自動的に読み込まれる。一方、電源をオフする時は M P 2 0 によりキャッシュメモリ 7 内のアドレステーブルを、読み込んできたドライブ 1 2 内の所定の場所に C P U 1 の閲知無しに自動的に格納する。

【0020】(読み出し処理) 次に、 A D C 2 内での具体的な I / O 処理について図 1 、図 2 を用いて説明する。 C P U 1 より発行されたコマンドは I F A d p 1 5 を介して A D C 2 に取り込まれ、 M P 2 0 により読み出し要求か書き込み要求か解読される。まず、読み出し要求の場合の処理方法を以下に示す。 M P 2 0 が読み出し要求のコマンドを認識すると、 M P 2 0 は C P U 1 から送られてきた論理アドレスをアドレステーブルを参照し、キャッシュメモリ 7 内に存在するかどうかキャッシュフラグ 3 2 を調べ、判定する。キャッシュフラグ 3 2 がオンでキャッシュメモリ 7 内に格納されている場合 (キャッシュヒット) は、 M P 2 0 がキャッシュメモリ 7 から当該データを読み出す制御を開始し、キャッシュメモリ 7 内に無い場合 (キャッシュミス) は当該ドライブ 1 2 へその内部の当該データを読みだす制御を開始する。キャッシュヒット時は M P 2 0 はアドレステーブルにより C P U 1 から指定してきた論理アドレス 2 7 に対し、当該データが格納されているキャッシュメモリ 7 のキャッシュアドレス 3 1 に変換し、キャッシュメモリ 7 へ当該データを読み出しに行く。具体的には M P 2 0 の指示の元でキャッシュアダプタ回路 (C A d p) 2 3 によりキャッシュメモリ 7 から当該データは読み出される。 C A d p 2 3 はキャッシュメモリ 7 に対するデータの読み出し、書き込みを M P 2 0 の指示で行う回路で、キャッシュメモリ 7 の状態の監視、各読み出し、書き込み要求に対し排他制御を行う回路である。 C A d p 2 3 により読み出されたデータはデータ制御回路 (D C C) 2 2 の制御によりチャネルインターフェース回路 (C H I F) 2 1 に転送される。 C H I F 2 1 では C P U 1 におけるチャネルインターフェースのプロトコルに変換し、チャネルインターフェースに対応する速度に速度調整する。具体的には C P U 1 、 A D C 2 間のチャネルインターフェースを光のインターフェースにした場合、光のインターフェースのプロトコルを A D C 2 内では電気処理でのプロトコルに変換する。 C H I F 2 1 におけるプロトコル変換および速度調整後は、チャネルバスディレクタ 5 において、チャネルバススイッチ 1 6 が外部インターフェースバス 4 を選択し I F A d p 1 5 により C P U 1 へデータ転送を行なう。

【0021】一方、キャッシュミス時はキャッシュヒット時と同様にアドレステーブルにより、 C P U 1 が指定した論理アドレス 2 7 から当該ドライブ番号とそのドライブ 1 2 内の実際にデータが格納されている S C S I 内 A d d r 3 0 を認識し、そのアドレスに対し、 M P 2 0 は D r i v e I F 2 4 に対し、当該ドライブ 1 2 への読み出し要求を発行するように指示する。 D r i v e I F 2 4 では S C S I の読み出し処理手順に従って、読み出しコマンドをドライブユニットバス 9 - 1 または 9 - 2 を介して発行する。 D r i v e I F 2 4 から読み出しコマンドを発行された当該ドライブ 1 2 においては

指示されたSCSI内Addr30へシーク、回転待ちのアクセス処理を行なう。当該ドライブ12におけるアクセス処理が完了した後、当該ドライブ12は当該データを読み出しドライブユニットバス9を介してDriveIF24へ転送する。DriveIF24では転送されてきた当該データをドライブ12側のキャッシュアダプタ回路(CAdp)14に転送し、(CAdp)14ではキャッシュメモリ7にデータを格納する。この時、CAdp14はキャッシュメモリ7にデータを格納することをMP20に報告し、MP20はこの報告を元に、アドレステーブル内のCPUが読み出し要求を発行した論理アドレス27のキャッシュフラグ32をオン(1)にし、キャッシュアドレス31にキャッシュメモリ7内のデータを格納したアドレスを登録する。キャッシュメモリ7にデータを格納し、アドレステーブルのキャッシュフラグ32をオン(1)にし、キャッシュメモリ7内のアドレスを更新した後はキャッシュヒット時と同様な手順でキャッシュメモリ7からデータを読み出し、CPU1へ当該データを転送する。

【002.2】(書き込み処理)一方書き込み時は以下のように処理される。書き込み処理にはユーザが書き込み先の論理アドレスを指定し、そのデータを書き換える更新と、新たに空き領域に書き込む新規書き込みがある。CPU1から書き込み命令が発行されたとする。まず、ADC2のMP20はCPU1から書き込み要求のコマンドを受け取った後、コマンドを受け取ったMP20が所属するクラスタ13内の各チャネルバス6において処理可能かどうかを調べ、可能な場合は処理可能だという応答をCPU1へ返す。CPU1では処理可能だという応答を受け取った後にADC2へデータを転送する。この時、ADC2ではMP20の指示によりチャネルバスディレクタ5において、チャネルバススイッチ16が当該外部インターフェースバス4とIFAdp15を当該チャネルバス6と接続しCPU1とADC2間の接続を確立する。CPU1とADC2間の接続を確立後CPU1からのデータ転送を受け付ける。CPU1から転送されてくるデータには、論理アドレスと書き込みデータ(以下新データとする)があり、これらのデータはMP20の指示により、CHIF21によりプロトコル変換を行ない、外部インターフェースバス4での転送速度からADC2内の処理速度に速度調整する。CHIF21におけるプロトコル変換および速度制御の完了後、データはDCC22によるデータ転送制御を受け、CAdp24に転送され、CAdp23によりキャッシュメモリ7内に格納される。この時、CPU1から送られてきたデータが、論理アドレスの場合は、読み出しと同様にアドレステーブルによりアドレス変換を行い、物理アドレスに変換する。また、CPU1から送られてきたデータが新データの場合は、キャッシュメモリ7に格納したアドレスをアドレステーブル内のキャッシ

ュアドレス31に登録する。この時、書き込む新データをキャッシュメモリ7内に保持するときは、論理アドレス27のキャッシュフラグ32をオン(1)とし、保持しない場合はキャッシュフラグ32をオフ(0)とする。なお、キャッシュメモリ7内に保持されている新データに対し、さらに書き込み要求がCPU1から発行された場合は、キャッシュメモリ7内に保持されている新データを書き替える。

【002.3】キャッシュメモリ7に格納された新データは、この新データにより新しくパリティを更新し(以下更新されたパリティを新パリティとする)、以下のように論理グループ10内のドライブ12へ新データと新パリティを格納する。まず、すでにドライブ12内に書き込まれているデータを新しいデータに書き換える更新の場合についてのフローを図12を用いて示す。本発明ではパリティは論理グループ10を構成するドライブ12において、RAIDのレベル4のように特定のパリティ専用のドライブ12に格納する。

【002.4】本発明の書き込み処理方法を図3を用いて説明する。MP20はCPU1が指定した論理アドレスからアドレステーブルを参照し、データ、パリティが格納されているドライブ12(DDriveNo.29, PDriveNo.34で指定される)とそのドライブ12内の物理的なアドレスであるSCSI内Addr30, PSCI内Addr35を認識する。図3に示すようにCPU1からSD#1のドライブ12のData#1に対し、NewData#1に更新する書き込み要求が発行された場合、MP20はアドレステーブルにより更新されるデータ(旧データ)であるData#1および更新されるパリティ(旧パリティ)であるParity#1の物理アドレスを認識した後、それぞれのドライブに対し旧データと旧パリティの読み出しを行なう(図3、図12の(1))。この時の読み出し方法は先に説明した読み出し処理におけるドライブ12からキャッシュメモリ7への読み出しと同じである。ただ、書き込み時の読み出しでは、ADC2のMP20が発行した読み出し要求のため、読み出したデータはCPU1へは転送せず、キャッシュメモリ7に転送するのみである。この様に読み出した旧データ、旧パリティと書き込む新データとで排他的論理和を行ない更新後の新パリティであるNewParity#1を作成しキャッシュメモリ7に格納する(図3、図12の(2))。新パリティ(NewParity#1)のキャッシュメモリ7への格納完了後、MP20は新データ(NewData#1)をSD#1のドライブ12のData#1のアドレスに書き込む(図3、図12の(3))。なお、この新データの書き込みはMP20の管理の下で非同期に行なわれるようにもよい。新パリティ(NewParity#1)はキャッシュメモリ7にそのまま格納しておく。この時、図4に示すアドレステーブル

に対し論理アドレスが Data #1 のエントリに New Data #1 を登録し、キャッシュメモリ 7 に保持しておく場合はキャッシュアドレス 31 にキャッシュ内のアドレスを登録し、キャッシュフラグ 32 をオンとする。また、parity に関してはキャッシュメモリ 7 に保持したままのため、P キャッシュアドレス 36 にキャッシュアドレスを登録し、P キャッシュフラグ 37 をオンとする。なお、この様にアドレステーブルで P キャッシュフラグ 37 がオンとなっている parity は、更新済みの parity となり、parity 格納用のドライブ 1 2 内に格納されている parity は無効とされる。本発明では図 5 に示すように、CPU 1 からの新データを不揮発化されたキャッシュメモリ 7 内の領域に格納し、新 parity の作成が完了しキャッシュメモリ 7 に格納した時点で、MP 2 0 は書き込み処理を終了したと CPU 1 に報告する。なお、従来方法では図 5 に示したように、新 parity をドライブの 1 回転後に書き込み、MP 2 0 が書き込み処理を終了したと CPU 1 に報告している。新 parity のドライブ 1 2 への書き込みは MP 2 0 の管理の下で非同期に行なわれるため、ユーザからは見えない。また、新データのドライブ 1 2 への書き込みを MP 2 0 の管理の下に非同期に行なう場合は、同様にしてユーザからは見えない。以後 CPU 1 から Data #10, Data #8 に対する書き込み処理が発行されれば、上記と同様に処理し、各新 parity をキャッシュメモリ 7 に格納していく。

【0025】キャッシュメモリ 7 に溜められた新 parity は、予めユーザが設定した設定値以上の新 parity がキャッシュメモリ 7 に溜った場合か、または、ユーザからの読み出し／書き込み要求の発行されていないタイミングが生じた場合に parity 格納用のドライブ 1 2 にまとめて書き込む（図 3、図 12 の（4））。このように新 parity を parity 格納用のドライブ 1 2 にまとめて書き込む場合は、シーケンシャルに書き込まれる。この様に新 parity を parity 格納用のドライブ 1 2 に書き込む際に、アドレステーブルの PSCI 内 Addr 35 に実際に新 parity を書き込んだ SCS 1 内 Addr を登録する。従来方法では制御の簡略化のため、parity を格納するドライブ 1 2 内で parity を格納する PSCI 内 Addr 35 はデータを格納するドライブ 1 2 内のデータを格納する SCS 1 内 Addr 30 と同一にしていた。しかし、本発明では新 parity はシーケンシャルにまとめて書き込まれるため、parity 格納用のドライブ 1 2 内で parity を格納する PSCI 内 Addr 35 とデータ格納用のドライブ 1 2 内のデータを格納した SCS 1 内 Addr 30 は原則として同一にはせず、異なるものにしている。

【0026】また、キャッシュメモリ 7 内に溜められた新 parity をシーケンシャルにまとめて書き込む最中に CPU 1 より読み出し要求が発行された場合、読み出し

処理には parity は関与しないため、先に説明したように通常の読み出し処理を行なう。一方、キャッシュメモリ 7 内に溜められた新 parity をシーケンシャルにまとめて書き込む最中に CPU 1 より書き込み要求が発行された場合は、通常の書き込み処理と同様に旧データを読み出し、旧データの読み出し後新データを書き込む。この時、キャッシュメモリ 7 には新データと旧データを保持し、キャッシュメモリ 7 内に溜められた新 parity をシーケンシャルにまとめて書き込む処理が終了次第、当該書き込まれた新 parity を旧 parity として読み出し、この旧 parity と先の新データと旧データとにより新 parity を作成しキャッシュメモリ 7 に格納する。

【0027】本発明では新 parity をまとめ、シーケンシャルに書き込むが、図 7、8 に示すように、前に書き込まれている parity が有効な場合、その上に新 parity を書き込んで消すわけには行かない。本発明では、アドレステーブルで P キャッシュフラグ 37 がオンとなっている旧 parity は、更新済みの parity となり、parity 格納用のドライブ 1 2 内では無効となっているが、
20 P キャッシュフラグ 37 がオフとなっている parity は parity 格納用のドライブ 1 2 内においてまだ有効な parity である。この有効な parity が消されると、ドライブ障害が発生した場合、障害ドライブ内のこの parity の作成に関与したデータの回復が不可能となる。以下に新 parity のシーケンシャルまとめ書きの時の parity 格納用ドライブ 1 2 内の有効データの扱い方について説明する。図 7 に示すように parity 書き込み前のトラックにおいて parity の P1, P2, P3 はデータの書き込み要求に伴い更新された無効 parity であり、P
30 P8, P9 はデータに対する書き込み要求が発行されていないため、更新されていない有効 parity である。書き込み要求 1, 2, 3 の順に書き込み要求が発行され、これにより P2, P3, P1 の順に旧 parity が更新され更新済みの新 parity として P'2, P'3, P'1 の順にキャッシュメモリ 7 に格納されているとする。これらの新 parity を parity 格納用のドライブ 1 2 にシーケンシャルにまとめ書きする場合は、旧 parity P1 の位置に新 parity P'2 を書き込み、有効 parity である P8 はそのまま残し、旧 parity P2 の位置に新 parity P'3 を書き込み、旧 parity P3 の位置に新 parity P'1 を順に書き込んでいく。すなわち、旧 parity の一連の位置に新 parity をその書き込み要求の発行順に順次書き込むのである。以上のように本発明ではシーケンシャルにまとめ書きする際に、有効データはそのまま残し、飛ばして新 parity を書き込んでいく。なお、シーケンシャルに parity を書き込んでゆく際、1 つの parity を書き込み、次の parity の書き込みを開始するための処理をしている間に、次の parity を書き込むためのブロックが通り過ぎてしまわないように、
40 I-S ギャップ（これについては、“トランジスタ技術 S

P E C I A L N O. 27 第20頁”に記載されている)を充分に取る必要がある。また、別の方針としては図8に示すように有効パリティであるP8, P9をMP20の指示により擬似的な読み出し要求を発行し、この擬似的な読み出し要求によりキャッシュメモリ7に読み出し、この読み出しによりMP20はアドレステーブルのPキャッシュアドレス36をセットしPキャッシュフラグ37をオンとすることで、更新する新パリティとみなし、他の新パリティと一緒にシーケンシャルにまとめて書き込む方法もある。すなわち、図8に示すように、更新する新パリティP'2, P'3, P'1と更新する新パリティとみなしたパリティP8, P9からなる更新パリティ群の書き込み順をP'2, P'3, P'1, P8, P9とし、書き込み前のパリティ群P1, P8, P2, P3, P9の一連のアドレスへ上記書き込み順にしたがって更新パリティ群を順次書き込む。書き込み結果は図8のパリティ書き込み後のトラックに示すようになる。

【0028】一方ドライブ12にすでに格納されているデータに新しいデータ追加する新規書き込みの場合は、MP20はアドレステーブルにおいて空き領域を探す。空き領域には2種類ある。まず一つはまったく使用されていない未使用領域である。この様にまったく使用されていない領域では、アドレステーブルにおいて論理アドレス27の項に論理アドレスは登録されていない。このため、MP20はアドレステーブルにおいて論理アドレス27の項に論理アドレスが登録されていない領域を探すこと、未使用領域を見つけられる。もう一つの空き領域は、以前その領域は使用されていたが(データが書き込まれていた)、ユーザがそのデータが必要でなくなったため削除した削除領域である。削除領域は、アドレステーブルにおいて論理アドレス27の項に論理アドレスが登録されてるが無効フラグ28をオン(1)としている。このため、MP20はアドレステーブルにおいて無効フラグ28がオンになっている領域を探すこと、削除領域を見つけられる。MP20が新規書き込みを行なう空き領域を決定する場合、まず、未使用領域を探す。もし、未使用領域が無い場合は削除領域を新規書き込み先に使用する。これは、未使用領域はパリティの作成に関与していない(全て0で構成されているとした)ため、新規書き込みの際のパリティの更新は、新規書き込みする新データと更新される旧パリティとの排他的論理和のみで行なえるが、削除領域のデータはユーザにとっては意味が無いデータとなっているが、パリティの作成には関与しているため、新規書き込みの際に旧データと同じように読み出して、旧パリティと新規書き込みデータとの間で排他的論理和をとり新パリティを作成しなければならない。このため、未使用領域に新規書き込みを行なうのと、削除領域に新規書き込みを行なうのでは、削除領域から削除されたデータを読み出す処理が入

らない分、未使用領域に新規書き込みを行なう方が早く処理できるためである。以上述べたようにMP20が空き領域を探し、空き領域の認識が完了した後、この空き領域に新規書き込みデータの書き込みを行ない、更新と同様にアドレステーブルに論理アドレス27を登録し、削除領域に新規書き込みを行なった場合は無効フラグ28をオフとする。以上述べたように、新規書き込みと更新では、新データの書き込み先が異なるのみで処理自体は同じである。

- 10 【0029】(障害回復処理) 次にドライブ12に障害が発生した場合の、障害ドライブ12内のデータを回復する方法を説明する。図3に示すようにSD#1のドライブ12のData#1とSD#2のドライブ12のData#2とSD#3のドライブ12のData#3とSD#4のドライブ12のData#4からSD#5のドライブ12のParity#1が作成されている。同様にData#5, 6, 7, 8からParity#2、Data#9, 10, 11, 12からParity#3が作成されている。SD#1, 2, 3, 4のドライブ12の中でどれか1台のドライブ12に障害が発生した場合、残りのドライブ12内のデータとパリティから、障害ドライブ12内のデータを回復する。本発明では、パリティはパリティを格納するドライブ12内においてランダムに格納されている。そこで、障害ドライブ12内のデータを回復する際は、MP20はSD#5のドライブ12内のパリティをキャッシュメモリ7内に全て読み出す。例えば、図3においてSD#1のドライブ12に障害が発生したとする。まず、MP20はSD#5のドライブ12からParity#1, 2, 3をキャッシュメモリ7に読み出し、アドレステーブルのPキャッシュアドレス36にキャッシュメモリ7内のアドレスを登録し、Pキャッシュフラグ37をオンにする。次に、MP20はSD#2, 3, 4のドライブ12からData#2, 3, 4をそれぞれ読み出し、これらのデータとアドレステーブルによりこれらのデータに対応するパリティを、アドレステーブルにより探し、当該パリティを見つけた後は、上記データと先に読み出している当該パリティであるParity#1とをパリティ生成回路(PG)25に送り、Data#1を復元する。同様にData#5, 9も復元する。この様に復元したデータは、障害ドライブ12を正常なドライブ12に交換した後、この正常なドライブ12に書き込むことで回復処理を行なう。また、ドライブ12の障害時に備え予め予備の正常なドライブ12を用意してある場合は、この予備の正常なドライブ12に復元したデータを書き込みことで回復処理を行なう。
- 【0030】以上の説明では更新後の新パリティを格納するキャッシュメモリ7は不揮発性半導体メモリとした。しかし、パリティはデータとは異なり停電等によつてキャッシュメモリ7から消失しても、新たに作り直す

ことが可能なため、この、新たに作成する手間を許容できるなら、キャッシングメモリ7内で旧パリティを格納する領域を揮発な半導体メモリにすることも可能である。以上の説明では、更新後の新パリティをキャッシングメモリ7に格納したが、キャッシングメモリ7ではなく専用のメモリを用意することも可能である。従来のレベル4, 5では書き込み処理を行なうたびに新パリティの書き込みを行なっていたため、常にパリティの更新後に回転の回転待ちを必要としたが、本発明ではシーケンシャルなまとめ書きを行なう際の最初に0, 5回転の回転待ちを必要とするのみである。

【0031】(実施例2) 本実施例では実施例1で示したように、1台のパリティ格納用のドライブ12にシーケンシャルにまとめ書きするのではなく、複数のパリティ格納用のドライブ12に対し新パリティをパラレルに書き込む方法を示す。本実施例でも実施例1と同じ処理により、データの書き込みに伴いパリティを更新し、更新した新パリティはキャッシングメモリ7に保持される。図9に示すように書き込み要求1, 2, 3によりData#1, #9, #8がそれぞれNew Data#1, #9, #8に更新され、このデータの更新により更新された新パリティとしてNew Parity#1, #3, #2がキャッシングメモリ7に保持されている(図9の(1) (2) (3))。実施例1と同様に予めユーザの設定値以上の新パリティがキャッシングメモリ7に溜った場合か、または、ユーザからの読み出し/書き込み要求の発行されていないタイミングで、複数のパリティ格納用のドライブ12であるSD#5, SD#6にパラレルにまとめて書き込む(図9の(5))。パラレルにまとめて書き込む単位としては、レベル3のようにバイト単位と、レベル4, 5のようにパリティ単位がある。この時、各パリティ格納用のドライブ12に対する書き込み方法は、実施例1の1台のパリティ格納用のドライブ12への書き込み方法と同じである。また、本実施例の変形として、パラレルに書き込む新パリティによりパリティを作成し、SD#7のパリティ格納用のドライブ12に書き込む。この様にパリティのパリティを作成することにより、パリティ格納用のドライブ12の障害時に新たにパリティを作成する際に、データを読み出す必要が無く、その間のデータへのアクセスを受け付けることが可能となる。

【0032】(実施例3) 本実施例では、図6に示すようにパリティ格納用のドライブ12を複数の領域に分割し、各領域単位で行なう方法を説明する。この領域の分割は、SCSI内Addr30により行なう。例えばSCSI内Addr30がDADR1からDADRkまでを領域1とする。SD#1, 2, 3, 4のドライブ12においてSCSI内Addr30がDADR1からDADRkまで領域1に所属する各データに対するパリティは、SD#5のドライブ12のP SCSI内Addr3

6がDADR1からDADRkの領域1に格納される。このように、アドレステーブルにおいて、データ、パリティに対し所属する領域を対応させる。この様な領域分割を行なった場合、領域1に所属するパリティに対しデータの書き込みによるパリティの更新が行なわれた場合、新パリティは領域1のパリティとしてキャッシングメモリ7に保持する。同様にCPU1からの他の書き込みによる新パリティをキャッシングメモリ7に保持している、領域1のパリティとして保持されている新パリティはまとめられ、領域1にシーケンシャルにまとめて書き込む。他の領域に対しても同様に各領域に所属するパリティの新パリティは、まとめてそれぞれの領域にシーケンシャルに書き込む。また、この様に領域に分割した際のドライブ12に障害が発生した場合の回復方法は、基本的には実施例1で示した領域に分割しない場合と同じである。異なるのは、障害ドライブ12内のデータを回復する際に、MP20はSD#5のドライブ12内のパリティをキャッシングメモリ7内に全て読み出さず、各領域単位に読み出す。つまり、回復処理を領域単位で行なう。

【0033】本実施例の変形例を以下に示す。本変形例では、図13に示すように、パリティを書き込む領域を、1台のパリティ書き込み用の専用ドライブ12に限定せず、論理グループ10を構成するドライブ12全体に分散させる。この様に、パリティを書き込む領域を論理グループ10を構成するドライブ12に分散させた場合と上記のように1台のドライブ12に限定した場合で異なる点を以下に示す。上記のように1台のドライブ12に限定した場合、パリティの格納先のドライブ12が限定されているため、MP20がアドレステーブルによりパリティが格納されている領域を決定する場合、SCSI内Addrのみで可能である。一方、本変形例のように、パリティを書き込む領域を論理グループ10を構成するドライブ12に分散させた場合、MP20がアドレステーブルにより領域を決定する場合、SCSI内Addrの他にドライブ番号も必要となる。この様に本変形例ではアドレス変換方法が異なるが、その他の制御方法は同じである。

【0034】(実施例4) 以下本発明の実施例4を説明する。図14は、本実施例のハードウェア構成を示し、1はCPU、2はディスクアレイコントローラ(以下ADC)、3はディスクアレイユニット(以下ADU)である。ADU3は複数の論理グループ10により構成され、個々の論理グループ10はm台のドライブ12とフラッシュメモリコントローラ(FMEMC)42と複数のフラッシュメモリチップ(FMEMチップ)40により構成されるフラッシュメモリ(FMEM)41と、各々のドライブ12またはFMEM41とADC2を接続するディスクアレイユニットバス9-1から9-4により構成される。本実施例では、各論理グループ1

0内にパリティを専用に格納するF M E M 4 1を設け、書き込み時にA D C 2がレベル4の制御により作成したパリティをA D C 2内のキャッシュメモリ7に溜め、これらのパリティを一度にシーケンシャルにF M E M 4 1に書き込む所に特徴がある。このように作成したパリティを溜め、まとめて一度にシーケンシャルにF M E M 4 1に格納することで、ドライブ1 2で構成されたレベル4のディスクアレイで問題となった書き込み時のパリティ更新オーバヘッド（回転待ち時間）を削減することが可能になる。なお、このドライブ1 2の数については特に制限は無い。この論理グループ1 0は障害回復単位で、この論理グループ1 0内の各ドライブ1 2内の各データによりパリティを作成する。本実施例ではA D C 2のM P 2 0はm台の個々のドライブ1 2内のデータから各々のパリティが作成され、これらのパリティが論理グループ1 0の一ヶ所に集めて格納されるレベル4の制御を行う。従来のディスクアレイではこのパリティはドライブに格納されていた。

【0035】次にA D C 2の内部構造についての説明であるが、これは実施例1における図1および図2を用いた説明と同様であるので省略する。なお、実施例1の図2に対応する本実施例の図15ではパリティ格納用のディスクドライブがフラッシュメモリ(F M E M)4 1に置き換えられている。また、(アドレス変換法)については、実施例1における説明と同様であるので省略する。但し、実施例1の図4におけるP D r i v e N o. 3 4、P S C S I内Add r 3 5は図4に対応する本実施例の図17においてはF M E M 3 4'、F M E M内Add r 3 5'となっている。次に、(読み出し処理)についての説明は、実施例1における説明と同様であるので省略する。

【0036】(書き込み処理) 次に本実施例の特徴となる書き込み時の処理について以下に示す。書き込み処理にはユーザが書き込み先の論理アドレスを指定し、そのデータを書き換える更新と、新たに空き領域に書き込む新規書き込みがある。C P U 1から書き込み命令が発行されたとする。まず、A D C 2のM P 2 0はC P U 1から書き込み要求のコマンドを受け取った後、コマンドを受け取ったM P 2 0が所属するクラスタ1 3内の各チャネルバス6において処理可能かどうかを調べ、可能な場合は処理可能だという応答をC P U 1へ返す。C P U 1では処理可能だという応答を受け取った後にA D C 2へデータを転送する。この時、A D C 2ではM P 2 0の指示によりチャネルバスディレクタ5において、チャネルバススイッチ1 6が当該外部インターフェースバス4とI F A d p 1 5を当該チャネルバス6と接続しC P U 1とA D C 2間の接続を確立する。C P U 1とA D C 2間の接続を確立後C P U 1からのデータ転送を受け付ける。C P U 1から転送されてくるデータには、論理アドレスと書き込みデータ(以下新データとする)があり、

これらのデータはM P 2 0の指示により、C H I F 2 1によりプロトコル変換を行ない、外部インターフェースバス4での転送速度からA D C 2内での処理速度に速度調整する。C H I F 2 1におけるプロトコル変換および速度制御の完了後、データはD C C 2 2によるデータ転送制御を受け、C A d p 2 4に転送され、C A d p 2 3によりキャッシュメモリ7内に格納される。この時、C P U 1から送られてきたデータが、論理アドレスの場合は、読み出しと同様にアドレステーブルにより10アドレス変換を行い、物理アドレスに変換する。また、C P U 1から送られてきたデータが新データの場合は、キャッシュメモリ7に格納したアドレスをアドレステーブル内のキャッシュアドレス3 1に登録する。この時、書き込む新データをキャッシュメモリ7内に保持するときは、論理アドレス2 7のキャッシュフラグ3 2をオン(1)とし、保持しない場合はキャッシュフラグ3 2をオフ(0)とする。なお、キャッシュメモリ7内に保持されている新データに対し、さらに書き込み要求がC P U 1から発行された場合は、キャッシュメモリ7内に保持20されている新データを書き替える。

【0037】キャッシュメモリ7に格納された新データは、この新データにより新しくパリティを更新し(以下更新されたパリティを新パリティとする)、以下のように論理グループ1 0内のドライブ1 2へ新データを格納し、F M E M 4 1に新パリティを格納する。まず、すでにドライブ1 2内に書き込まれているデータを新しいデータに書き換える更新の場合についてのフローを図12を用いて示す。本実施例では論理グループ1 0において、R A I Dのレベル4の制御を行い、パリティはパリティ専用のF M E M 4 1に格納する。

【0038】本実施例の書き込み処理方法を図16を用いて説明する。M P 2 0はC P U 1が指定した論理アドレスからアドレステーブルを参照し、データが格納されているドライブ1 2(D D r i v e N o. 2 9で指定される)とそのドライブ1 2内の物理的なアドレスであるS C S I内Add r 3 0とパリティが格納されているF M E Mチップ4 0のアドレスであるF M E Mアドレス3 4'、このF M E Mチップ0内の物理アドレスであるF M E M内Add r 3 5'を認識する。図16に40示すようにC P U 1からS D # 1のドライブ1 2のData # 1に対し、New Data # 1に更新する書き込み要求が発行された場合、M P 2 0はアドレステーブルにより更新されるデータ(旧データ)であるData # 1および更新されるパリティ(旧パリティ)であるParity # 1の物理アドレスを認識した後、旧データの格納されているドライブ1 2と旧パリティの格納されているF M E Mチップ4 0に対し読み出しを行なう(図16、図12の(1))。この時の旧データの読み出し方法は先に説明した読み出し処理におけるドライブ1 2からキャッシュメモリ7への読み出しと同じであ

る。F MEM 4 1 からの旧パリティの読み出しが以下のように行う。MP 2 0 は図 1 7 に示すアドレステーブルを参照し、更新される旧パリティのPアドレス 3 3 に対応する、旧パリティが格納されている物理アドレス(F MEMアドレス 3 4')と、このF MEMアドレス 3 4' のF MEMチップ 4 0 内のF MEM内Add r 3 5')を認識する。MP 2 0 がこのように更新される旧パリティのF MEM 4 1 内の物理アドレスを認識した後は、F MEM 4 1 内のF MEMC 4 2 に対し、当該旧パリティの読み出しコマンドとこの物理アドレスを送る。読み出しコマンドと物理アドレスを受け取ったF MEMC 4 2 では、F MEMアドレス 3 4' に対するF MEMチップ 4 0 をイネーブルにし、F MEM内Add r 3 5' を当該F MEMチップ 4 0 にセットし、当該旧パリティを読み出し、F MEMC 4 2 内のバッファに格納する。当該旧パリティを読み出したF MEMC 4 2 では、MP 2 0 に対し当該旧パリティの格納されている当該F MEMチップ 4 0 からの読み出しが完了したことを報告する。この報告を受け取ったMP 2 0 はF MEMC 4 2 に対し、当該旧パリティをキャッシュメモリ 7 に転送するように指示し、この指示を受け取ったF MEMC 4 2 は当該旧パリティをキャッシュメモリ 7 に転送する。ただ、書き込み時の読み出しへは、ADC 2 のMP 2 0 が発行した読み出し要求のため、読み出したデータはCPU 1 へは転送せず、キャッシュメモリ 7 に転送するのみである。この様に読み出した旧データ、旧パリティと書き込む新データとで排他的論理和を行ない更新後の新パリティであるNew Parity # 1 を作成しキャッシュメモリ 7 に格納する(図 1 6、図 1 2 の(2))。新パリティ(New Parity # 1)のキャッシュメモリ 7 への格納完了後、MP 2 0 は新データ(New Data # 1)をSD # 1 のドライブ 1 2 のData # 1 のアドレスに書き込む(図 1 6、図 1 2 の(3))。なお、この新データの書き込みはMP 2 0 の管理の下で非同期に行なわれるようにしてよい。新パリティ(New Parity # 1)はキャッシュメモリ 7 にそのまま格納しておく。この時、図 1 7 に示すアドレステーブルに対し論理アドレスがData # 1 のエントリにNew Data # 1 を登録し、キャッシュメモリ 7 に保持しておく場合はキャッシュアドレス 3 1 にキャッシュ内のアドレスを登録し、キャッシュフラグ 3 2 をオンとする。また、パリティに関してはキャッシュメモリ 7 に保持したままのため、Pキャッシュアドレス 3 6 にキャッシュアドレスを登録し、Pキャッシュフラグ 3 7 をオンとする。なお、この様にアドレステーブルでPキャッシュフラグ 3 7 がオンとなっているパリティは、更新済みのパリティとなり、F MEM 4 1 内に格納されているパリティは無効とされる。本実施例では図 5 に示すように、CPU 1 からの新データを不揮発化されたキャッシュメモリ 7 内の領域に格納し、新パリティ

の作成が完了しキャッシュメモリ7に格納した時点でのMP20は書き込み処理を終了したとCPU1に報告する。なお、従来方法では図5に示したように、新parityをドライブの1回転後に書き込み、MP20が書き込み処理を終了したとCPU1に報告している。新parityのFMEM41への書き込みはMP20の管理の下で非同期に行なわれるため、ユーザからは見えない。また、新データのドライブ12への書き込みをMP20の管理の下に非同期に行なう場合は、同様にしてユーザからは見えない。以後CPU1からData#10、Data#8に対する書き込み処理が発行されれば、上記と同様に処理し、各新parityをキャッシュメモリ7に格納していく。

【0039】キャッシュメモリ7に溜められた新parityは、予めユーザが設定した設定値以上の新parityがキャッシュメモリ7に溜った場合か、または、ユーザからの読み出し／書き込み要求の発行されていないタイミングが生じた場合にparity格納用のFMEM41にまとめて書き込む(図16、図12の(4))。このように新parityをparity格納用のFMEM41にまとめて書き込む場合は、シーケンシャルに書き込まれる。また、この様に新parityをparity格納用のFMEM41に書き込む際に、アドレステーブルのFMEMアドレス34'、FMEM内Addr35'に実際に新parityを書き込んだアドレスを登録する。従来のディスクアレイは全てドライブで構成され、データもparityもドライブに格納されていた。また、新データは旧データが格納されていたアドレスに書き戻されるため、更新による書き込み時にアドレステーブルの変更を必要とせず、制御は簡単であった。しかし、本実施例では、新parityは書き込み要求が発行された順にシーケンシャルにまとめて書き込まれるため、parity格納用のFMEM41内では、新parityが格納されるアドレス(FMEMアドレス34'とFMEM内Addr35')は原則として、その新parityに対する旧parityが格納されていたアドレスとは同一にせず、異なるものにしている。

【0040】また、キャッシュメモリ7内に溜められた新parityをシーケンシャルにまとめて書き込む最中にCPU1より読み出し要求が発行された場合、読み出し処理にはparityは関与しないため、先に説明したように通常の読み出し処理を行なう。一方、キャッシュメモリ7内に溜められた新parityをシーケンシャルにまとめて書き込む最中にCPU1より書き込み要求が発行された場合は、通常の書き込み処理と同様に旧データを読み出し、旧データの読み出し後新データを書き込む。この時、キャッシュメモリ7には新データと旧データを保持し、キャッシュメモリ7内に溜められた新parityをシーケンシャルにまとめて書き込む処理が終了次第、当該書き込まれた新parityとで新たにparityを更新する。

【0041】本実施例では新パリティをまとめ、シーケンシャルに書き込むが、図19、20に示すように、前に書き込まれているパリティが有効な場合、その上に新パリティを書き込んで消すわけには行かない。本実施例では、アドレステーブルでPキャッシュフラグ37がオンとなっている旧パリティは、更新済みのパリティとなり、パリティ格納用のFMEM41内では無効となっているが、Pキャッシュフラグ37がオフとなっているパリティはFMEM41内においてまだ有効なパリティである。この有効なパリティが消されると、ドライブ障害が発生した場合、障害ドライブ内のこのパリティの作成に関与したデータの回復が不可能となる。

【0042】そこで、新パリティをシーケンシャルにまとめ書きする時のパリティ格納用FMEM41内の有効データの扱い方について以下に説明する。図19に示すようにパリティ書き込み前のFMEMチップ40においてパリティのP1、P2、P3はデータの書き込み要求に伴い更新された無効パリティであり、P8、P9はデータに対する書き込み要求が発行されていないため、更新されていない有効パリティである。書き込み要求1、2、3の順に書き込み要求が発行され、これによりP2、P3、P1の順に旧パリティが更新され更新済みの新パリティとしてP'2、P'3、P'1の順にキャッシュメモリ7に格納されているとする。これらの新パリティをパリティ格納用のFMEM41にシーケンシャルにまとめ書きする場合は、MP20は書き込む新パリティの量を調べ、この量に相当すFMEMチップ40内の無効パリティ（P1、P2、P3）が書き込まれているアドレスを一度に消去する（フラッシュメモリの書き込みにおける消去方法についてはHN28F1600シリーズのデータシート（ADJ-203-045

(a) (z)に記載されている）。このように、新パリティをシーケンシャルにまとめ書きするFMEMチップ40内の消去が完了すると、旧パリティP1のアドレスに新パリティP'2を書き込み、有効パリティであるP8はそのまま残し、旧パリティP2の位置に新パリティP'3を書き込み、旧パリティP3の位置に新パリティP'1を順に書き込んでいく。すなわち、旧パリティの一連の位置に新パリティをその書き込み要求の発行順に順次書き込むのである。以上のように本発明ではシーケンシャルにまとめ書きする際に、有効データをそのまま残し、飛ばして新パリティを書き込んでいく。

【0043】また、別の方法としては図20に示すように有効パリティであるP8、P9をMP20の指示により、FMEM41に対し擬似的な読み出し要求を発行し、この擬似的な読み出し要求によりキャッシュメモリ7に有効パリティを読み出し、この有効パリティのキャッシュメモリ7への読み出しによりMP20はアドレステーブルのPキャッシュアドレス36をセットしPキャッシュフラグ37をオンとすることで、更新する新パリ

ティとみなす。MP20はキャッシュメモリ7に溜められている新パリティと読み出された有効パリティの量を調べ、この量に相当するFMEMチップ40内の領域を一度に消去する。このように、FMEMチップ40内の消去が完了すると、FMEM41から読み出した有効パリティは新パリティとして、他の新パリティと一緒にシーケンシャルにまとめて書き込む。すなわち、図20に示すように、更新する新パリティP'2、P'3、P'1と更新する新パリティとみなしたパリティP8、P9からなる更新パリティ群の書き込み順をP'2、P'3、P'1、P8、P2、P3、P9とし、書き込み前のパリティ群P1、P8、P2、P3、P9の一連のアドレスへ上記書き込み順にしたがって更新パリティ群を順次書き込む。書き込み結果は図20のパリティ書き込み後のトランクに示すようになる。フラッシュメモリではデータをフラッシュメモリに書き込む場合、まず書き込むアドレスに格納されているデータを消去し、この消去が完了した後に実際にデータを書き込む。フラッシュメモリでは1セクタ（フラッシュメモリをアクセスする場合、ディスクにアクセスするときのアドレスと同じフォーマットのアドレスでアクセスする）を消去しても、複数セクタを一度に消去しても消去時間は同じである。又、書き込み時間の大半はこの消去時間が占めており、実際にフラッシュメモリに書き込む時間は消去時間と比較した場合無視できるほど小さい。本実施例の特徴であるシーケンシャルにまとめ書き込みを行うことにより、消去が一回で済むため、新パリティのまとめる数が多いほどオーバヘッドが小さく出来る。

【0044】本実施例では、FMEM41を構成する全FMEMチップ40に対し、平均に書き込みが行われるようにFMEMチップ40への書き込み回数をカウントする。フラッシュメモリはドライブと比較してランダムにアクセスしても短時間で処理できる利点があるが、その反面書き込み回数に制限がある。このため、あるアドレスに集中して書き込みが行われると、フラッシュメモリの一部が書き込み回数の限界に達してしまい、以後書き込めなくなってしまう。本実施例では、FMEM41に対しシーケンシャルにまとめ新パリティを書き込むため、FMEM41へ新パリティをシーケンシャルにまとめ書き込む際に、規則的に書き込むことで全FMEMチップ40に対し書き込み回数を平均化する。具体的には、図18に示すようにFMEM41がFMEM#1、2、3、4、nのn個のFMEMチップ40で構成され、アドレスが0000からffffとする。MP20は新パリティのシーケンシャルなまとめ書きを行うことを認識すると、前回のまとめ書きを行った際に、新パリティを格納した最終のアドレスを調べる。例えば、前回のまとめ書きにおいて、FMEM41内で0000からaaaaまで新パリティを書き込んだとすると、アドレスaaaaをMP20は記憶してお

く。そして、次のシーケンシャルなまとめ書き時にはこの記憶しておいたアドレス (aaa) を調べる。このように、MP20が前回の最終アドレスを記憶すると、MP20は次の新パリティのシーケンシャルなまとめ書きはアドレスaaaの次から行うように判断する。MP20がこのようにして新パリティのシーケンシャルなまとめ書きを行う先頭アドレスを認識すると、次にFMEM41への書き込み回数の限界に達しているかを判定する。

【0045】図22の書き込み回数判定フローチャートに示すように、新パリティをシーケンシャルにまとめ書きする先頭アドレスが0000かを判定する(51)。0000でなければ判定フローは終了し(52)、0000であればMP20は書き込み回数カウンタのカウンタ値に1をたす(53)。すなわち、先頭アドレスに来る度にカウンタの値を増加する。次にMP20はこの1をたしたカウンタ値が、予めセットされているFMEMチップ40の書き込み回数の限界値かを判定する(54)。FMEMチップ40の書き込み回数の限界値は、初期設定において予めMP20に対しユーザが設定する。この判定結果において、FMEMチップ40の書き込み回数の限界値を越えていなければ判定フローを終了し(52)、越えている場合はFMEMチップ40の交換を通報する(55)。以上のように本実施例ではFMEM41に対する新パリティのシーケンシャルなまとめ書きは、アドレスの低い方から高い方へ一方向に行われる。これにより、FMEM41の全FMEMチップ40では、書き込み回数は平均化されることになる。本実施例ではこのようにしてFMEM41内においてFMEMチップ40への書き込み回数を平均化する。

【0046】一方ドライブ12にすでに格納されているデータに新しいデータ追加する新規書き込みの場合は、MP20はアドレステーブルにおいて空き領域を探す。空き領域には2種類ある。まず一つはまったく使用されていない未使用領域である。この様にまったく使用されていない領域では、アドレステーブルにおいて論理アドレス27の項に論理アドレスは登録されていない。そのため、MP20はアドレステーブルにおいて論理アドレス27の項に論理アドレスが登録されていない領域を探すこと、未使用領域を見つけられる。もう一つの空き領域は、以前その領域は使用されていたが（データが書き込まれていた）、ユーザがそのデータが必要でなくなったため削除した削除領域である。削除領域は、アドレステーブルにおいて論理アドレス27の項に論理アドレスが登録されてるが無効フラグ28をオン（1）としている。このため、MP20はアドレステーブルにおいて無効フラグ28がオンになっている領域を探すこと、削除領域を見つけられる。MP20が新規書き込みを行なう空き領域を決定する場合、まず、未使用領域を探

す。もし、未使用領域が無い場合は削除領域を新規書き込み先に使用する。これは、未使用領域はパリティの作成に関与していない（全て0で構成されているとした）ため、新規書き込みの際のパリティの更新は、新規書き込みする新データと更新される旧パリティとの排他的論理和のみで行なえるが、削除領域のデータはユーザにとっては意味が無いデータとなっているが、パリティの作成には関与しているため、新規書き込みの際に旧データと同じように読み出して、旧パリティと新規書き込みデータとの間で排他的論理和をとり新パリティを作成しなければならない。このため、未使用領域に新規書き込みを行なうと、削除領域に新規書き込みを行なうのでは、削除領域から削除されたデータを読み出す処理が入らない分、未使用領域に新規書き込みを行なう方が早く処理できるためである。以上述べたようにMP20が空き領域を探し、空き領域の認識が完了した後、この空き領域に新規書き込みデータの書き込みを行ない、更新と同様にアドレステーブルに論理アドレス27を登録し、削除領域に新規書き込みを行なった場合は無効フラグ28をオフとする。以上述べたように、新規書き込みと更新では、新データの書き込み先が異なるのみで処理自体は同じである。

【0047】（障害回復処理）次にドライブ12に障害が発生した場合の、障害ドライブ12内のデータを回復する方法を説明する。図16に示すようにSD#1のドライブ12のData#1とSD#2のドライブ12のData#2とSD#3のドライブ12のData#3とSD#4のドライブ12のData#4からFMEM#1のFMEMチップ40のParity#1が作成されている。同様にData#5、6、7、8からParity#2、Data#9、10、11、12からParity#3が作成されている。SD#1、2、3、4のドライブ12の中でどれか1台のドライブ12に障害が発生した場合、残りのドライブ12内のデータとFMEM41内のパリティから、障害ドライブ12内のデータを回復する。本実施例では、パリティはパリティを格納するFMEM41内においてランダムなアドレスに格納されている。例えば、図16においてSD#1のドライブ12に障害が発生し、このSD#1のドライブ12内のData#1に読み出し要求が発生したとする。まず、MP20はアドレステーブルから読み出し要求が発行されたData#1は障害が発生したドライブ内のデータだと認識した場合は、SD#2、3、4のドライブ12からData#2、3、4をそれぞれ読み出し、MP20はアドレステーブルによりこれらのデータに対応するパリティを、アドレステーブルにより探す。MP20がアドレステーブルにより当該パリティのFMEM41内のアドレスを認識した後はFMEM41から当該パリティであるParity#1を読み出し、上記データと共にパリティ生成回路（PG）25に送り、排他的論

理和を行うことで Data # 1 を復元する。同様に Data # 5、# 9 も復元する。フラッシュメモリはドライブのようにディスクを回転させたり、ヘッドをシークさせるような機械的な動作を必要としない。また、フラッシュメモリでは先に述べたように書き込み時は消去を必要とするが、読み出し時には DRAM のような半導体メモリからの読み出しと同じように短時間で読み出せる。このようにフラッシュメモリにおけるランダムな読み出しはドライブと比較して無視できるほど短時間で処理できる。このため、本実施例のようにデータの回復毎にアドレステーブルを調査し、アドレスを認識し、FMEM 4 1 から読み出しても、ドライブからのデータの読み出しと比較し短時間で処理できる。この様に復元したデータは、障害ドライブ 1 2 を正常なドライブ 1 2 に交換した後、この正常なドライブ 1 2 に書き込むことで回復処理を行なう。また、ドライブ 1 2 の障害時に備え予備の正常なドライブ 1 2 を用意してある場合は、この予備の正常なドライブ 1 2 に復元したデータを書き込みことで回復処理を行なう。

【0048】以上の説明では更新後の新パリティを格納するキャッシュメモリ 7 は不揮発な半導体メモリとした。しかし、パリティはデータとは異なり停電等によってキャッシュメモリ 7 から消失しても、新たに作り直すことが可能なため、この、新たに作成する手間を許容できるなら、キャッシュメモリ 7 内で旧パリティを格納する領域を揮発性半導体メモリにすることも可能である。以上の説明では、更新後の新パリティをキャッシュメモリ 7 に格納したが、キャッシュメモリ 7 ではなく専用のメモリを用意することも可能である。従来のレベル 4、5 では新パリティをドライブに格納しているため、書き込み処理を行なうたびに新パリティの書き込みを行なっていたため、新パリティを書き込む毎に回転の回転待ちを必要としたが、本実施例ではシーケンシャルなまとめ書きを行なう際の FMEM チップ 4 0 の一括消去（約 10ms）と書き込みを行う時間のみである。

【0049】（実施例 5）本実施例では実施例 4 で示したように、1 つのパリティ格納用の FMEM チップにシーケンシャルにまとめ書きするのではなく、複数のパリティ格納用の FMEM チップ 4 0 に対し新パリティをパラレルに書き込む方法を示す。本実施例でも実施例 4 と同じ処理により、データの書き込みに伴いパリティを更新し、更新した新パリティはキャッシュメモリ 7 に保持される。図 2 1 に示すように書き込み要求 1、2、3 により Data # 1、# 9、# 8 がそれぞれ New Data # 1、# 9、# 8 に更新され、このデータの更新により更新された新パリティとして New Parity # 1、# 3、# 2 がキャッシュメモリ 7 に保持されている（図 2 1 の（1）（2）（3））。実施例 1 と同様に予めユーザの設定値以上の新パリティがキャッシュメモリ 7 に溜った場合か、または、ユーザからの読み出し

／書き込み要求の発行されていないタイミングで、複数のパリティ格納用の FMEM 4 1 である FMEM # 1、FMEM # 2 にパラレルにまとめて書き込む（図 2 1 の（5））。パラレルにまとめて書き込む単位としては、レベル 3 のようにバイト単位と、レベル 4、5 のようにパリティ単位がある。この時、各パリティ格納用の FMEM 4 1 に対する書き込み方法は、実施例 1 のパリティ格納用の FMEM 4 1 への書き込み方法と同じである。また、本実施例の変形として、パラレルに書き込む新パリティによりパリティを作成し、FMEM # 3 のパリティ格納用の FMEM 4 1 に書き込む。この様にパリティのパリティを作成することにより、パリティ格納用の FMEM 4 1 の障害時に新たにパリティを作成する際に、データを読み出す必要が無く、その間のデータへのアクセスを受け付けることが可能となる。

【0050】

【発明の効果】従来はデータの書き込みによるパリティの更新ごとにドライブに書き込むため、この更新された新パリティの書き込みに 1 回転の回転待ちが必要であった。

20 本発明によれば、更新した新パリティをキャッシュメモリに貯蔵しておき、後に、この様に貯蔵された新パリティをまとめてシーケンシャルにパリティ格納用のドライブに書き込んでいるため、シーケンシャルにまとめ書きする際の、最初に 0、5 回転の回転待ちが必要になるのみであり、従来問題となっていた書き込み時の処理時間を減少させることが可能となる。また、更新した新パリティをキャッシュメモリに貯蔵しておき、後に、この様に貯蔵された新パリティをまとめてシーケンシャルにパリティ格納用の FMEM に書き込んでいるため、シーケンシャルにまとめ書きする際の、FMEM の消去時間と書き込み時間のオーバヘッドが必要になるのみであり、書き込み時の処理時間を減少させることが可能となる。また、パリティを格納する FMEM の書き込み回数の平均化が可能になる。

【図面の簡単な説明】

【図 1】実施例 1 のハードウェア構成を示す図である。

【図 2】図 1 のチャネルバスディレクタと 1 クラスタの内部構造を示した図である。

【図 3】実施例 1 の書き込み処理時におけるデータ移動を説明する図である。

【図 4】アドレス変換テーブルを説明する図である。

【図 5】書き込み処理のタイミングチャートを示す図である。

【図 6】磁気ディスク装置内の領域分割を説明する図である。

【図 7】新パリティのシーケンシャル書き込み方法を説明する図である。

【図 8】新パリティの他のシーケンシャル書き込み方法を説明するである。

50 【図 9】実施例 2 の書き込み処理時におけるデータ移動

を説明する図である。

【図10】RAIDのレベル4、5における更新処理を説明する図である。

【図11】磁気ディスク装置内のアドレスを説明する図である。

【図12】データおよびパリティの書き込み処理のフローチャートを示す図である。

【図13】実施例3における磁気ディスク装置内の領域分割を説明する図である。

【図14】実施例4のハードウェア構成を示す図である。

【図15】図14のチャネルパスディレクタと1クラス
タの内部構造を示した図である。

【図16】実施例4の書き込み処理時におけるデータ移動を説明する図である。

【図17】実施例4におけるアドレス変換テーブルを説明する図である。

【図18】FMEM内のアドレスを説明する図である。
【図19】実施例4における新parityのシーケンシャー

ル書き込み方法を説明する図である。

【図20】実施例4における新パーティの他のシーケン

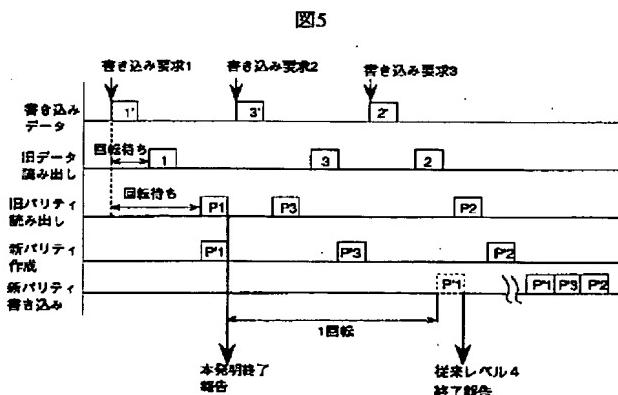
シャル書き込み方法を説明する図である。
【図21】実施例5の書き込み処理時におけるデータ移

【図22】FMEMの書き込み回数を判定するフロー

【篇目の説明】

- 1 CPU
 - 2 ディスクアレイコントローラ (ADC)
 - 3 ディスクアレイユニット (ADU)
 - 4 外部インターフェースバス
 - 5. チャネルバスディレクタ
 - 6 チャネルバス
 - 7 キッシュュメモリ

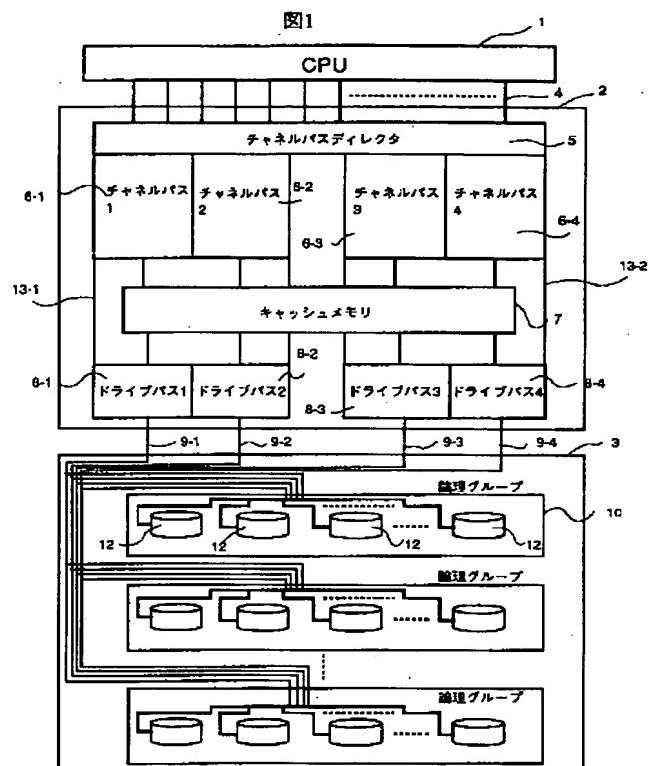
〔図5〕



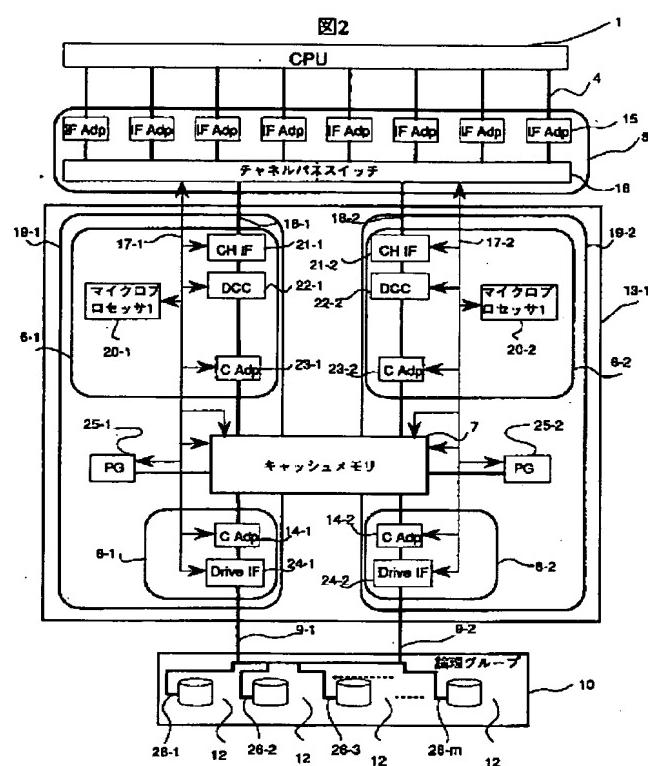
- 8 ドライブパス
 9 ディスクアレイユニットバス
 10 論理グループ
 12 ドライブ
 13 クラスタ
 14 ドライブ側キャッシュアダプタ (C Ad p)
 15 インターフェースアダプタ
 16 チャネルバススイッチ
 17 制御信号線
 10 18 データ線
 19 パス
 20 マイクロプロセッサ (M P)
 21 チャネルインターフェース回路 (C H I F)
 22 データ制御回路 (D C C)
 23 チャネル側キャッシュアダプタ (C Ad p)
 24 ドライブインターフェース回路 (D r i v e I F)
 25 パリティ生成回路 (P G)
 27 論理アドレス
 20 28 無効フラグ
 29 データドライブ番号 (D D r i v e N o .)
 30 S C S I 内A d d r
 31 キャッシュアドレス
 32 キャッシュフラグ
 33 P論理アドレス
 34 パリティドライブ番号 (P D r i v e N o .)
 34 FMEMアドレス
 35 P S C S I 内A d d r
 35 FMEM内A d d r
 30 36 Pキャッシュアドレス
 37 Pキャッシュフラグ
 40 フラッシュメモリチップ (F M E Mチップ)
 41 フラッシュメモリ (F M E M)
 42 フラッシュメモリコントローラ (F M E M C)

【図17】

【図 1】



【図2】



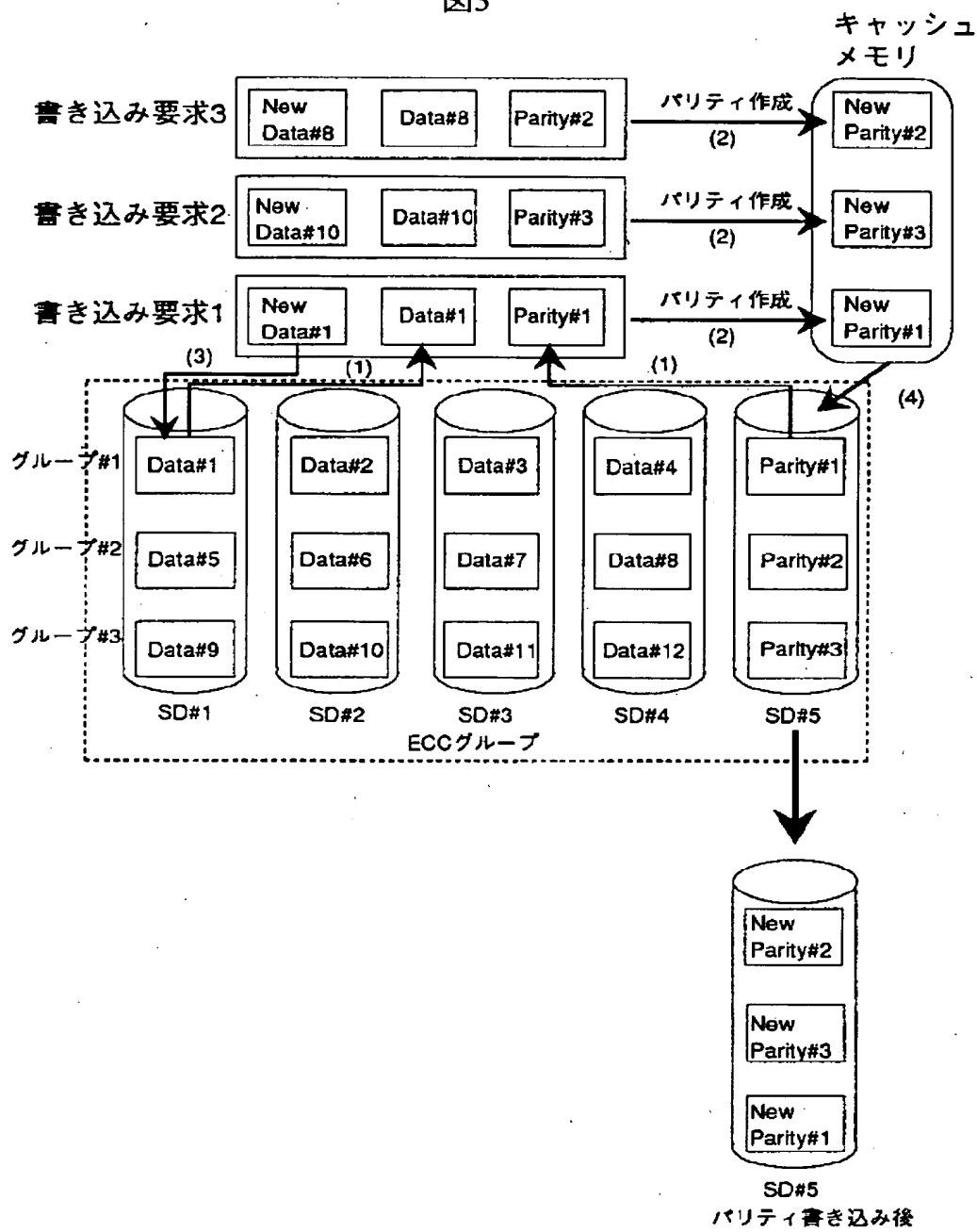
【図4】

四

論理 アドレス	無効 Flag	DDrive No.	SCSI 内Addr	キャッシュ アドレス	Cache Flag	P論理 アドレス	PDrive No.	PSCSI 内Addr	Pキャッシュ アドレス	PCache Flag
Data#1	0	SD#1	DADR1	—	0					
Data#2	0	SD#2	DADR1	C ADR5	1	Parity#1	SD#5	DADR5	C ADR21	1
Data#3	0	SD#3	DADR1	—	0					
Data#4	0	SD#4	DADR1	C ADR2	1					
Data#5	1	SD#1	DADR2	—	0					
Data#6	0	SD#2	DADR2	C ADR6	1	Parity#2	SD#5	DADR7	—	0
Data#7	0	SD#3	DADR2	C ADR1	1					
Data#8	0	SD#4	DADR2	—	0					
⋮	⋮	⋮	⋮	⋮	⋮			⋮	⋮	⋮

【図3】

図3



【図6】

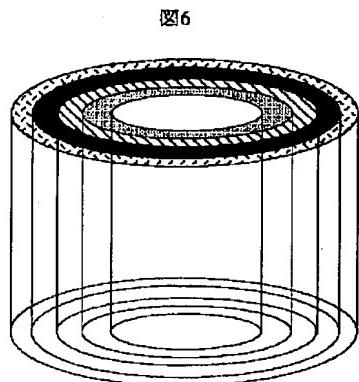


図6

【図7】

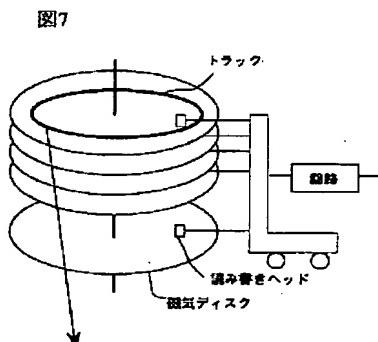


図7

【図8】

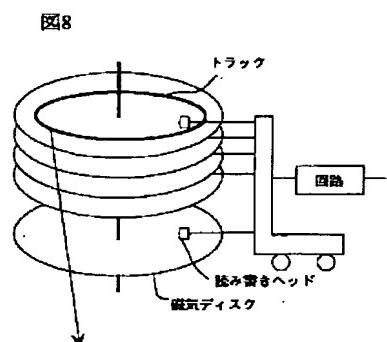
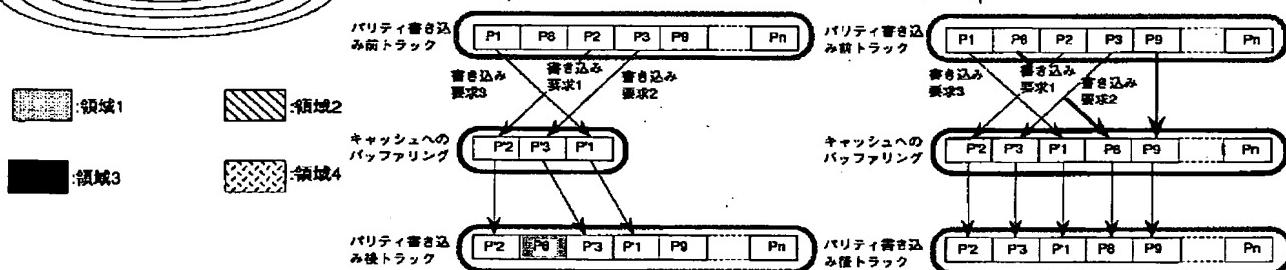
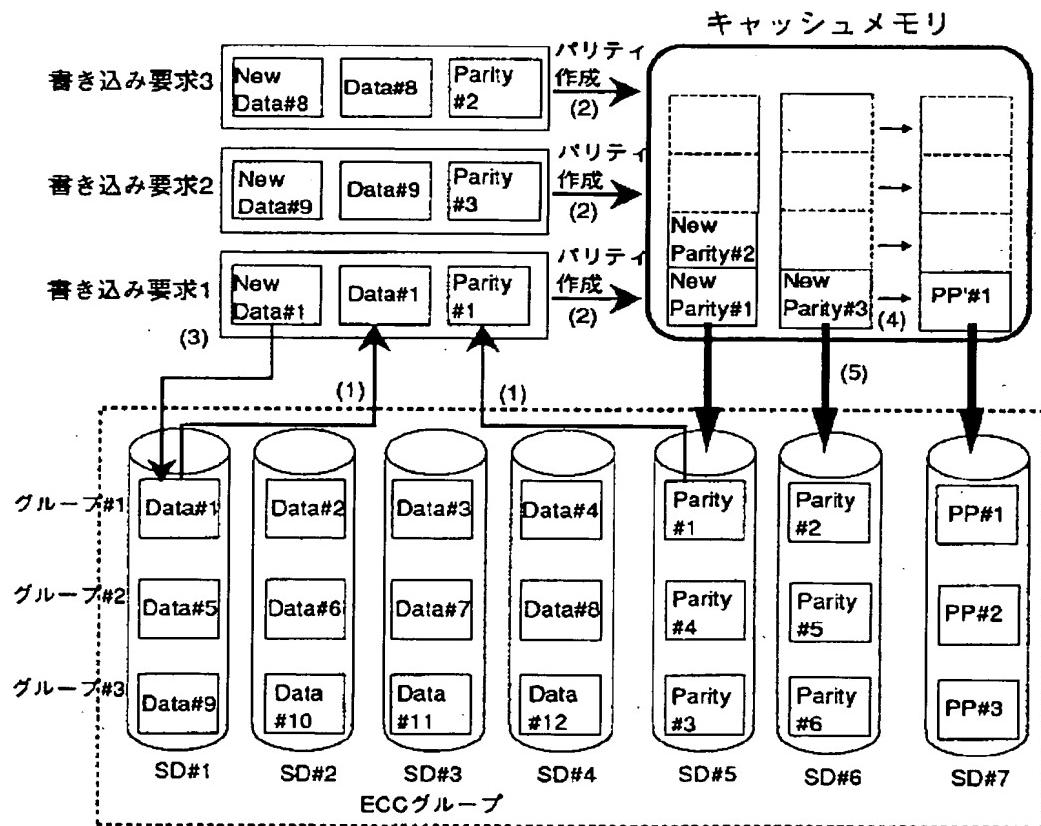


図8

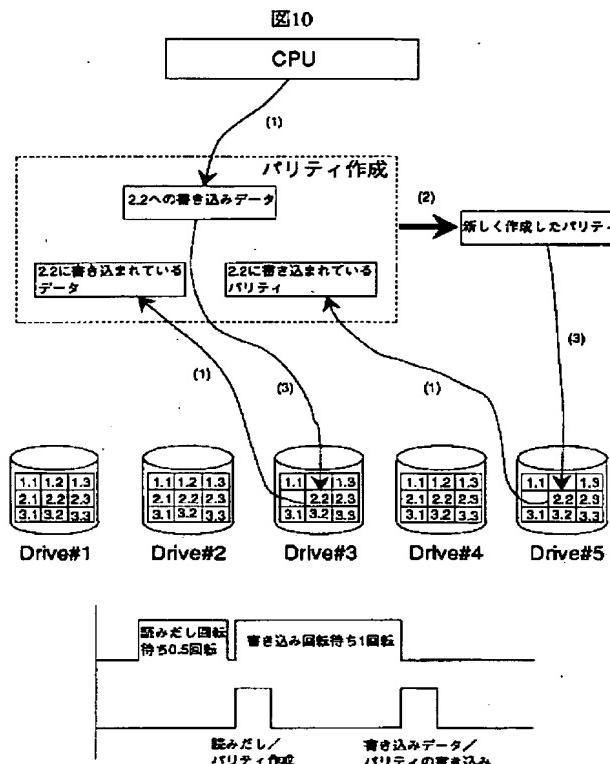


【図9】

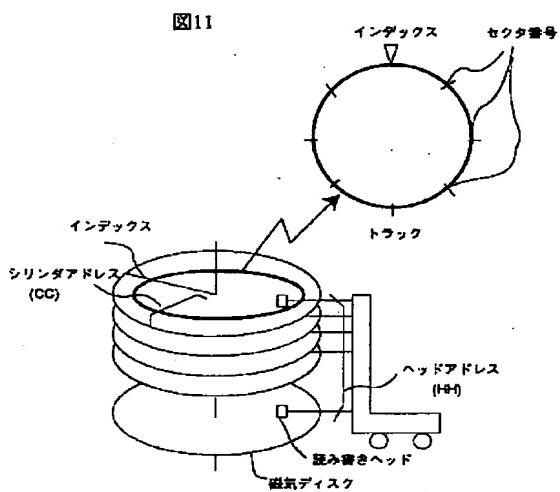
図9



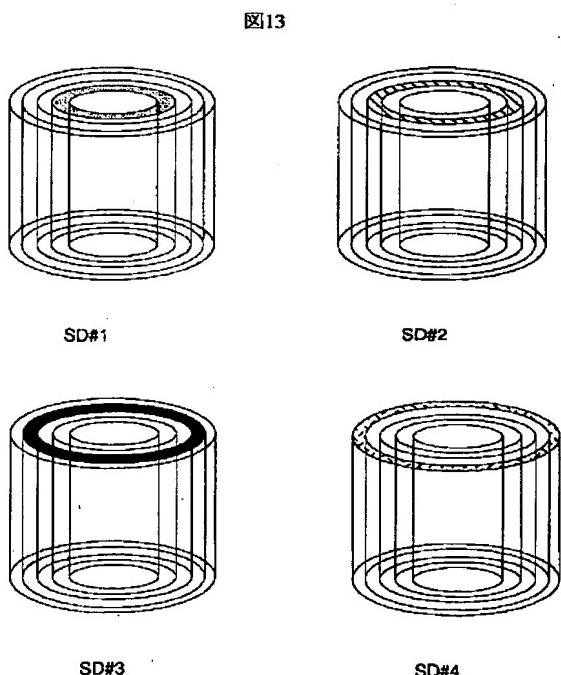
【図10】



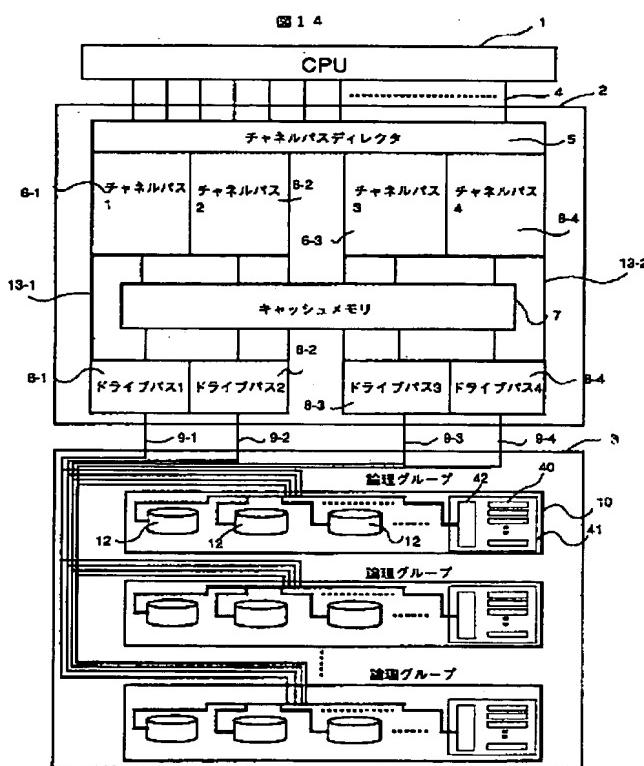
【図11】



【図13】

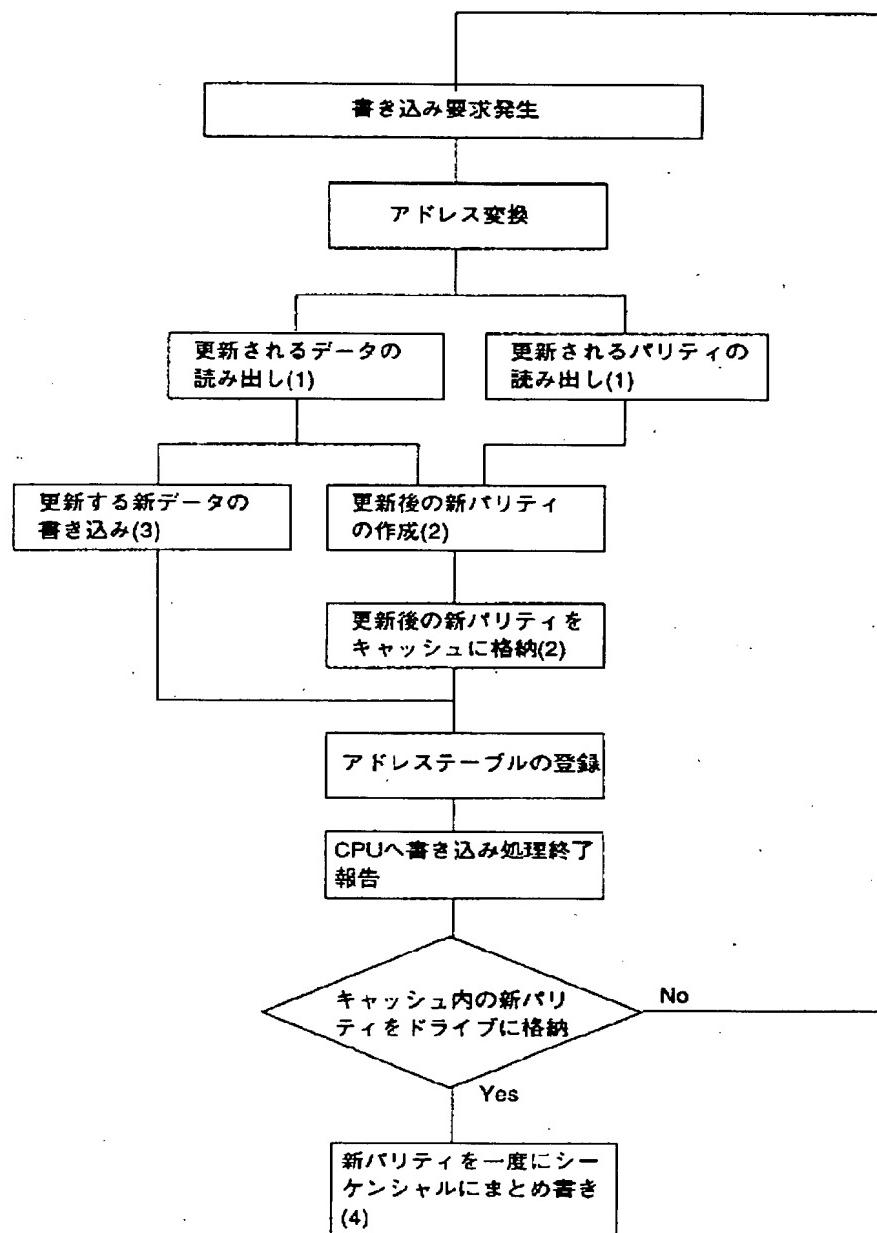


【図14】

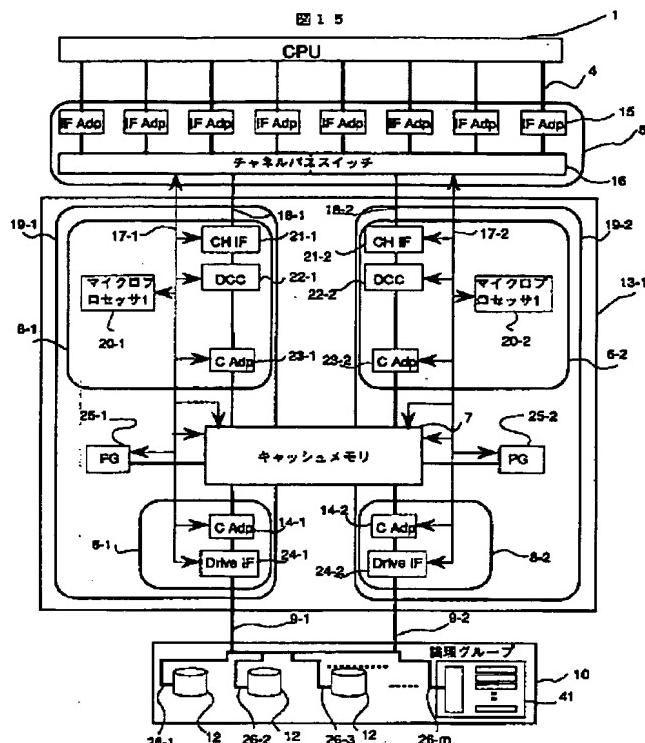


【図12】

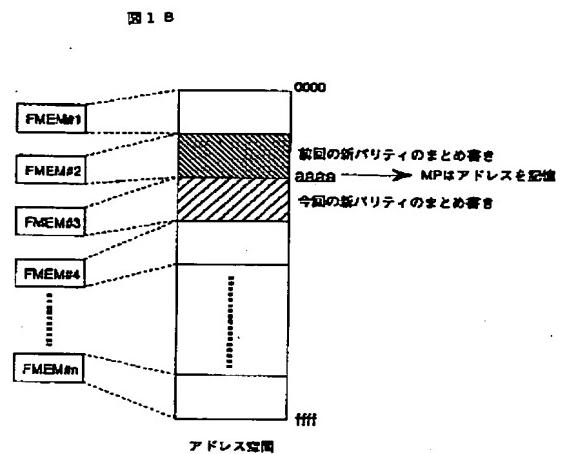
図12



【図15】

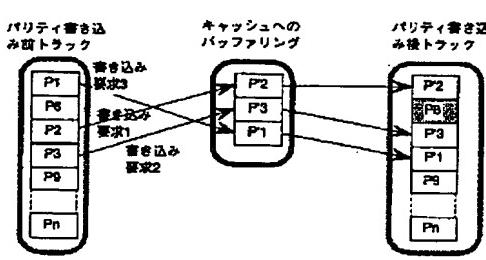


【図18】



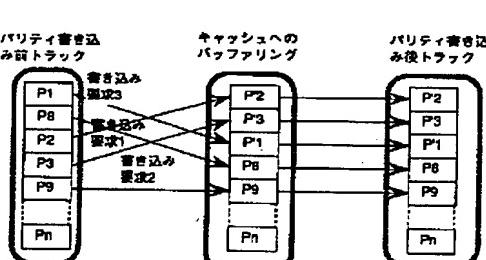
【図19】

図19



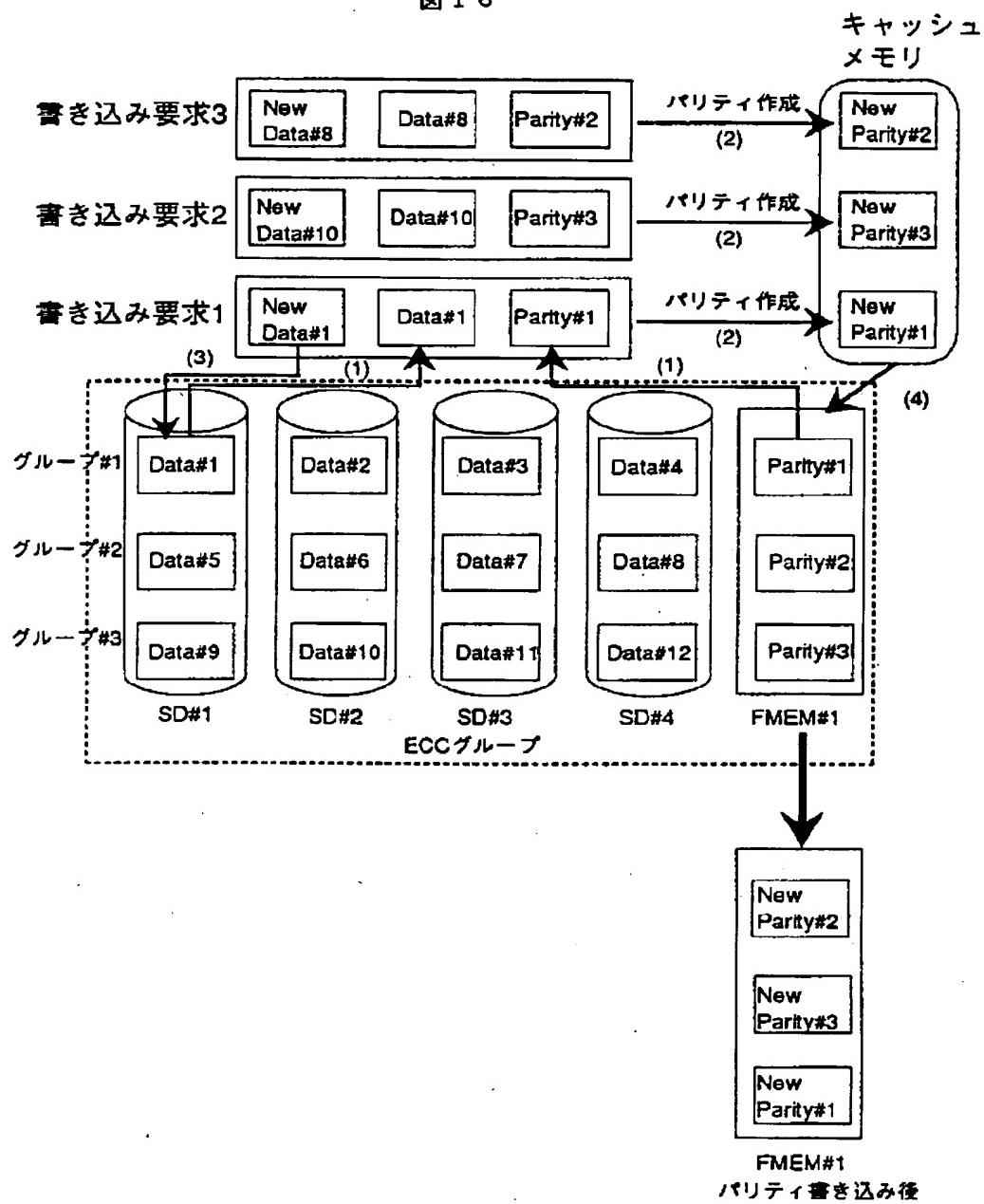
【図20】

図20



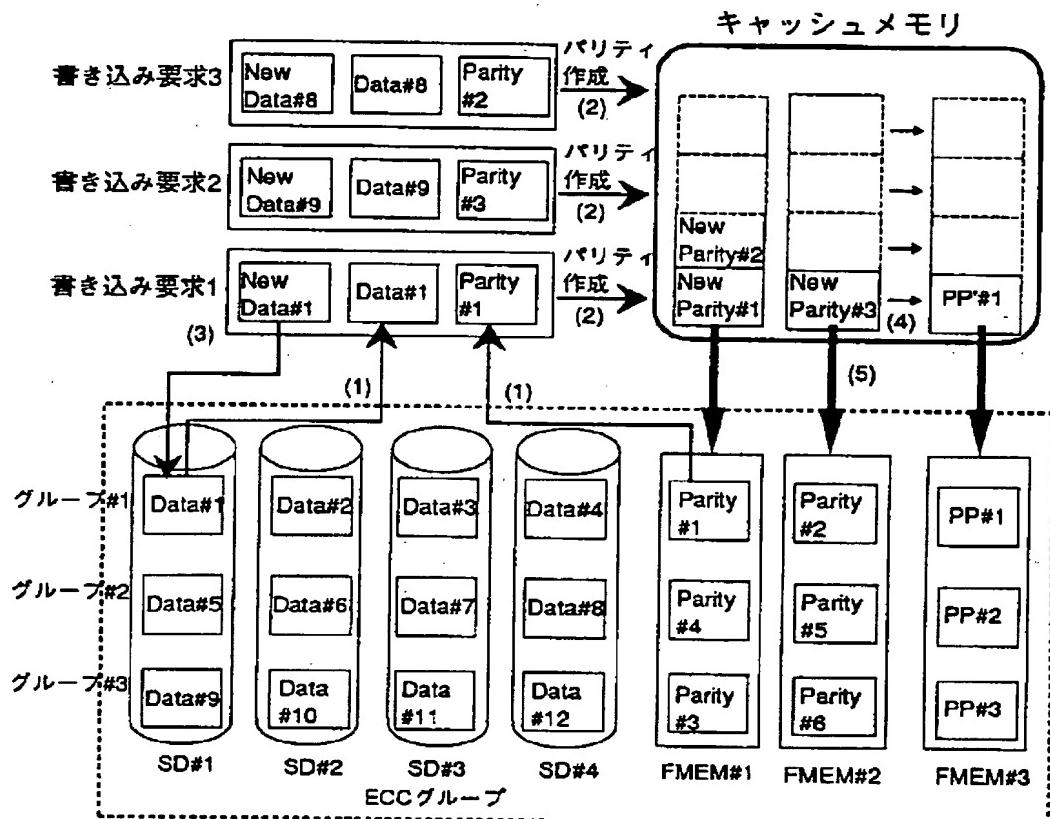
【図16】

図16



【図21】

図21



【図22】

図22

